N° d'enregistrement

au C.N.R.S.

THESE DE DOCTORAT D'ETAT ES SCIENCES MATHEMATIQUES

présentée

A L'UNIVERSITE PARIS VI

PAR

Madame.Christine.NORA.-.CHOUTEAU.....

pour obtenir
le grade de Docteur Es-Sciences

Sujet de la Thèse : "Une méthode de reconstitution et d'analyse de données incomplètes".

Soutenue le 17 FEV. 1975 devant la Commision d'Examen

MM..DUGUE..... Président
..BENZECRI..... Examinateur
..GEFFROY.....
..BELLIER.....
..DER.MEGREDITCHIAN....

Digitized by the Internet Archive
in 2026

https://archive.org/details/IA41548119_0009

° d'enregistrement

au C.N.R.S.

THESE DE DOCTORAT D'ETAT ES SCIENCES MATHEMATIQUES

présentée

A L'UNIVERSITE PARIS VI

PAR

Madame.Christine.NORA.-.CHOUTEAU.....

pour obtenir
le grade de Docteur Es-Sciences

Sujet de la Thèse : "Une méthode de reconstitution et d'analyse de données incomplèt

Soutenue le

devant la Commision d'Examen

MM..DUGUE..... Président
..BENZECRI..... Examinateur
..GE.FFROY.....
..BELLIER.....
..DER.MEGREDITCHIAN....

UNIV.
LEIDEN
BIBL.

I - SOMMAIRE

1°) Position du problème

Les statisticiens et notamment les utilisateurs de l'analyse des correspondances sont fréquemment confrontés au problème suivant : le tableau de leurs données est, pour une raison ou pour une autre, soit incomplet, soit entaché d'un certain nombre d'erreurs. De telles données traitées sans précaution ne conduiront probablement pas à des résultats satisfaisants.

Lorsqu'il est possible de réduire les ensembles de données étudiées sans pour autant affaiblir la compréhension de la correspondance, c'est-à-dire du phénomène analysé, la difficulté précédemment signalée peut être surmontée en supprimant les lignes ou les colonnes du tableau de données que l'on sait erronées - ou que l'on ne connaît pas- Une telle démarche revient cependant à abandonner une partie de l'information ou plutôt certains des paramètres ou aspects des problèmes étudiés : elle est bien évidemment inutilisable lorsque les ensembles de sujets et d'objets étudiés se trouvent nettement définis et ne peuvent être réduits sans nuire à la validité de l'étude, c'est-à-dire dans un grand nombre de cas.

L'objet de ce travail est donc de lever cette difficulté en tentant de remplacer les données manquantes ou que l'on sait erronées par des valeurs approchées permettant de rendre compte au mieux de la correspondance existant entre les ensembles étudiés.

Et ainsi que nous le verrons, les propriétés de l'analyse factorielle des correspondances permettent de déterminer un algorithme permettant sous certaines conditions de calculer, à partir des données connues, de telles valeurs approchées.

2°) Construction de l'algorithme TROU

Rappelons brièvement quelques propriétés de l'analyse des correspondances.

Soient I et J deux ensembles en correspondance et soit k(i,j) (ou indifféremment k_{ij}) l'élément en ligne i et colonne j du tableau k de la correspondance I x J, ou tableau k des données.

Notons k(i) et k(j) les sommes respectives des éléments de la ligne i et de la colonne j, k la somme de tous les éléments du tableau k.

En divisant tous les éléments $k(i,j)$ du tableau des données par k, nous pouvons normer le tableau k pour qu'il représente une mesure de probabilité f_{IJ} sur $I \times J$. Notons de même f_I et f_J les lois marginales sur I et J.

On considère alors dans $\mathbb{R}_J$ muni de la métrique du χ^2 de centre f_J le nuage $\mathcal{N}(I)$ des profils f_J^i affectés respectivement des masses f_i.

Soient $F_\alpha(i)$, $G_\alpha(j)$ les facteurs de ce nuage pour l'axe factoriel x correspondant à la valeur propre λ_α, différente de la valeur triviale 1. Soit A l'ensemble de ces axes factoriels.

Il est alors possible à partir de ces facteurs de reconstruire le tableau de données :

$$\forall (i,j) \in I \times J \qquad k(i,j) = \frac{k(i)\,k(j)}{k}\left[1 + \sum_{\alpha \in A} \lambda_\alpha^{-1/2}\, F_\alpha(i)\, G_\alpha(j)\right]$$

Supposons les valeurs propres rangées par ordre décroisssant, et gardons les r plus grandes valeurs propres seules ; il est alors possible d'obtenir une valeur approchée de $k(i,j)$:

$$k(i,j) \simeq \frac{k(i)\,k(j)}{k}\left[1 + \sum_{\alpha=1}^{r} \lambda_\alpha^{-1/2}\, F_\alpha(i)\, G_\alpha(j)\right] \qquad (1)$$

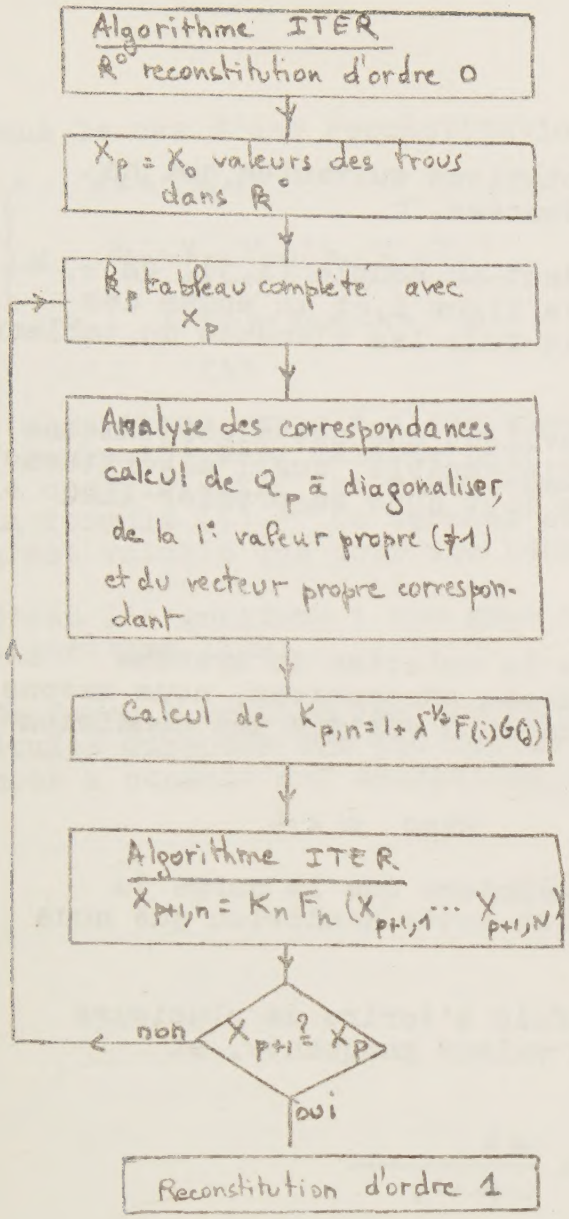
Le fait de se restreindre aux r plus grandes valeurs propres revient à considérer le nuage $\mathcal{N}'(I)$ d'un sous espace de dimension r, le plus proche de $\mathcal{N}(I)$ au sens des moindres carrés.

Cette formule approchée de reconstitution de données a fourni l'idée du présent algorithme de recherche de valeurs manquantes.

Introduisons quelques notations pour permettre une compréhension plus simple de l'algorithme TRØU.

Soit $\mathcal{M}$ l'ensemble des couples (i,j) correspondant aux valeurs manquantes du tableau k considéré.

Notons $x = (x_1, \ldots x_N)$ le vecteur colonne des valeurs calculées pour les N couples de $\mathcal{M}$ rangés dans un certain ordre fixé une fois pour toutes.



Représentons schématiquement cette recherche.

Le tableau initial $k_0 = k^0$ permet par la méthode d'analyse des correspondances de calculer la plus grande valeur propre (non triviale) et le vecteur propre correspondant.

Les nombres K_{on} se trouvent déterminés.

L'algorithme ITER permet alors de résoudre le système

$$\forall n = 1,..N \qquad X_{1n} = K_{on} F_n(X_{11} \cdots X_{1N})$$

Ce vecteur x_1 permet de reconstruire un tableau de données k_1 à partir duquel nous pouvons recommencer.

'algorithme s'arrêtera lorsque les vecteurs obtenus x_{p+1} et x_p seront les êmes.

ette méthode peut se généraliser pour obtenir une reconstitution d'ordre r, econstitution qui s'obtiendra, à partir de la reconstitution d'ordre r-1, en tilisant pour le calcul des nombres K_{pn} les r premiers facteurs.

°) Convergence de l'algorithme TROU

our la simplification des conditions trouvées, on a distingué quatre cas :

- une seule valeur manquante
- N valeurs manquantes dans des colonnes et des lignes toutes distinctes
- N valeurs manquantes dans une même colonne ou ligne
- N valeurs manquantes quelconques.

Dans la suite, nous utiliserons les notations suivantes qui ne dépendent que du tableau initial des données, k.

Pour une valeur manquante x correspondant au couple (i,j), on appellera a la somme des éléments de la ligne i, b la somme des éléments de la colonne j, c la somme de tous les éléments du tableau k.

D'après la construction, il est bien évident que cet algorithme ne pourra converger qu'à la condition indispensable que l'algorithme ITER soit lui-même convergent. Il nous faut donc en premier lieu étudier cet algorithme.

- Algorithme ITER

De façon générale, si nous recherchons la solution du système $x = G(x)$ par la méthode des approximations successives, nous savons que cet algorithme converge si la fonction G vérifie une condition de Lipschitz

$$\| G(x_1) - G(x_2) \| \leq \alpha \| x_1 - x_2 \| \qquad \text{avec} \quad \alpha < 1$$

En fait, pour faciliter l'étude, on demandera que la norme du jacobien de G soit inférieure à 1. C'est cette condition que nous avons utilisée pour le cas général.

De plus, un système $x = G(x)$ peut parfois s'écrire de plusieurs façons, par exemple dans le cas d'une valeur manquante, et d'une reconstitution d'ordre 0 :

$$x = \frac{(a+x)(b+x)}{c+x} \qquad \Longleftrightarrow \qquad x = \frac{a\,b}{c-(a+b)}$$

La première formule donnera lieu à un algorithme réel convergent ; dans le cas de la deuxième formule, l'algorithme se réduira au calcul d'un seul nombre.

Cette remarque nous a conduit à définir deux algorithmes.

En effet, considérons le système : $\forall n \in [1,N]$

$$X_n = K_n F_n(X_1 \cdots X_N) = K_n \frac{(a_n + X_n + Y_n)(b_n + X_n + Z_n)}{c + X_n + Y_n + Z_n + T_n}$$

où Y_n est la somme des autres valeurs manquantes de la même ligne i, Z_n de la même colonne j, T_n ni de la même ligne ni de la même colonne.

Dans le cas d'une reconstitution d'ordre 0, nous obtenons :

$$\begin{cases} \forall n \in [1,N] \\ X_n = \dfrac{(a_n + X_n + Y_n)(b_n + X_n + Z_n)}{c + X_n + Y_n + Z_n + T_n} \end{cases} \quad (3) \qquad \Longleftrightarrow \qquad \begin{cases} \forall n \in [1,N] \\ X_n = \dfrac{(a_n + Y_n)(b_n + Z_n)}{c - (a+b) + T_n} \end{cases} \quad (4)$$

L'algorithme, appelé algorithme 1, basé sur l'utilisation de la formule (3) sera valable pour toutes reconstitutions (en rétablissant le coefficient K_n si $r > 0$). Un deuxième appelé algorithme 2 utilisera la formule (4). Ce dernier est plus rapide que l'algorithme 1 mais n'est valable que pour une reconstitution d'ordre 0.

C'est l'algorithme 1 que nous utiliserons de façon générale comme algorithme ITER;

De façon à réduire les conditions suffisantes de convergence, des études directes ont été faites pour les cas particuliers. Ceci nous a conduit aux conditions suivantes (cf. chapitre IV).

	Algorithme 1	Algorithme 2
1 trou	Convergence si $K < \frac{c^2}{\left[\sqrt{a(c-b)} + \sqrt{b(c-a)}\right]^2}$	1 seul calcul $\frac{ab}{c-(a+b)}$
N trous dans des lignes et colonnes distinctes i.e. aucune ligne ni colonne ne comporte plus d'un trou	Convergence si $\forall n \in [1,N]$ $c - K_n(a_n+b_n) \geq a_n b_n K_n \frac{(2K_n+N-3)^2}{K_n+N-2}$	Convergence si $\sum_{n=1}^{N} \frac{a_n b_n}{\left[c-(a_n+b_n)\right]^2} < 1 + \min_{n\in[1,N]} \frac{a_n b_n}{\left[c-(a_n+b_n)\right]^2}$
N trous même ligne ou même colonne	Convergence si $\forall n \in [1,N]$ $\sqrt{a\left(K_n c - \sum_{\ell=1}^{N} K_\ell b_\ell\right)} + \sqrt{\sum_{\ell=1}^{N} K_\ell b_\ell (c-a)} \leq c$	Convergence si $\sum_{n=1}^{N} \frac{b_n}{c-(a_n+b_n)} < 1 + \min_{n\in[1,N]} \frac{b_n}{c-(a+b_n)}$
N quelconques	Convergence si $\forall n \in [1,N]$ $K_n\left[(1+N_y)\frac{b_n+X_o+Z_o}{c+X_o+Z_o} + (1+N_z)\frac{a_n+X_o+Y_o}{c+X_o+Y_o} + (2N_t-N)\frac{(a_n+X_o+Y_o)(b_n+X_o+Z_o)}{(c+X_o+Y_o)(c+X_o+Z_o)}\right] < 1$ Si $2N_t - N < 0$ remplacer le dernier terme par $(2N_t-N)\frac{a_n b_n}{(c+T_o)^2}$	Convergence si $\forall n \in [1,N]$ $N_y \frac{b_n+Z_o}{c-a_n-b_n} + N_z \frac{a_n+Y_o}{c-a_n-b_n} + N_t \frac{(b_n+Z_o)(a_n$

X_o majoration de la valeur du $n^{ème}$ trou
Y_o majoration de la somme des N_y valeurs des trous dans la même ligne que le $n^{ème}$ tr
Z_o majoration de la somme des N_z valeurs des trous dans la même colonne que le $n^{ème}$
T_o majoration de la somme des N_t valeurs des trous ni dans la même ligne ni dans la même colonne que le $n^{ème}$ trou.

On voit déjà que les conditions suffisantes de convergence de l'algorithme 2 sont moins restrictives que celles de l'Algorithme 1.

Il faut remarquer que quel que soit l'algorithme qu'on veuille utiliser comme algorithme ITER, la convergence ne pourra avoir lieu que s'il existe des solutions au système considéré. En particulier, si l'existence d'une solution est certaine, il sera possible de changer cet algorithme i pour un autre, comme celui de Newton-Raphson, aux conditions moins restrictives mais beaucoup plus lourd au point de vue du calcul puisqu'il exige à chaque pas la résolution d'un système linéaire.

Pour étudier cette existence, nous avons étudié deux algorithmes nommés 3 et 4 (l'un correspondant à l'algorithme 2, l'autre à l'algorithme 1) opérant par récurrence sur le nombre n de trous : Pour n trous, on fait varier alternativement les valeurs de (n-1) trous, celle du $n^{ième}$ trou, celles de (n-1) trous ....

Le tableau suivant donne les résultats trouvés dans les cas particuliers considérés.

	Algorithme 3	Algorithme 4
N trous dans des lignes et colonnes distinctes	solution si $\forall n \in [1,N]$ $\frac{a_n b_n}{[c-(a_n+b_n)]^2} < 1$	solution si $\forall n \in [1,N]$ $K_n < \frac{c+\sqrt{a_n b_n}}{a_n+b_n+2\sqrt{a_n b_n}}$
N trous dans une même ligne ou colonne	toujours une solution	solution assurée si $K_n < 1$

Ces deux derniers algorithmes n'ont pu être écrit en FORTRAN.

- algorithme TROU

L'étude de la convergence de l'algorithme complet nous a amené à résoudre les problèmes suivants (cf chapitre V).

- variations de la matrice Q à diagonaliser en fonction des variations des valeurs manquantes
- variations des valeurs propres et vecteurs propres de Q_0 en fonction des variations de Q.
- majoration de la norme du jacobien du système $x_n = K_n F_n(x_1 \dots x_N)$

En conclusion, ces études conduisent à un système de trois conditions suffisantes de convergence de l'algorithme TROU, de la forme :

$$\begin{cases} - \; C \, \|X_2 - X_1\|_1 \leq \delta/8 \\ - \; M \sum_{p=1}^{r} \left\{ \frac{C}{2\sqrt{\lambda_{1p} - \delta/4}} \sum_{n=1}^{N} \left[(|\varphi_{1n}^p| + 2\frac{A_J}{\delta_p} \|X_2 - X_1\|_1)(|\varphi_{1n}^p| + \frac{2A_I}{\delta_p} \|X_2 - X\|_1) \right] \right. \\ \left. + \; \frac{\sqrt{\lambda_{1p} + \delta/4}}{\delta_p} \left[A_J \sum_{n=1}^{N} (|\varphi_{1n}^p| + \frac{2A_I}{\delta_p} \|X_2 - X_1\|_1) + A_I \sum_{n=1}^{N} (|\varphi_{1n}^p| + \frac{2A_I}{\delta_p} \|X_2 - X_1\|_1) \right] \right\} \leq \frac{1}{2} \\ - \text{ conditions de convergence de l'algorithme ITER.} \end{cases}$$

Les paramètres introduits ne dépendent tous que des éléments du tableau initial, des valeurs de départ de l'algorithme Trou (X_1) ainsi que des valeurs obtenues après la première itération (X_2).

En effet, l'étude des vecteurs propres ne peut avoir lieu que si les variations des valeurs propres restent limitées.

Dans les cas particuliers considérés, on a pu déterminer exactement, sous certaines conditions restrictives, l'inverse du jacobien du système $x = G(x)$.

Ces conditions restrictives restent légères à savoir :

- $x_n < \sqrt{a_n b_n}$ dans le cas de lignes et colonnes distinctes

- $\begin{cases} x_n < a_n \\ \sum_{n=1} x_n < b \end{cases}$ dans le cas d'une même colonne.

On a pu donner sous forme plus simplifiée la valeur de certains paramètres mais ceci n'a été possible qu'en rendant plus restrictives les conditions suffisantes trouvées.

Si nous recherchons "la" meilleure reconstitution du tableau, et non plus la meilleure reconstitution d'ordre r, la convergence de l'algorithme TROU pour tout r assure la convergence de l'algorithme MU.

D'après le calcul des K_n celle-ci aura lieu au pire pour r = inf (card I, card J) - 1.

En fait, les vecteurs propres étant normés et les valeurs propres rangées par ordre décroissant, il est bien évident qu'on pourra s'arrêter lorsque les variations sur K seront inférieures à la précision souhaitée.

4°) Résultats

- Reconstitution d'ordre 0

Dans ce cas, l'algorithme TROU se réduit à l'algorithme ITER et les algorithmes 1 et 2 ont pu être utilisés. La seule différence obtenue consiste en un nombre d'itérations moindre pour l'algorithme 2, aussi ne distinguerons nous pas ces 2 algorithmes.

D'autre part, il est possible de changer les sommes partielles intervenant dans les formules après le calcul de chaque trou ou au contraire lorsque tous les trous ont été calculés. Les résultats étant identiques, nous ne ferons à nouveau aucune différence.

La recherche d'une reconstitution d'ordre 0 a été faite pour trois tableaux différents, appelés MU, CI, CA, sur chacun de ces tableaux, on a tiré au hasard des couples (i,j) correspondant aux données manquantes ; on a reconstitué ces derniers par le programme puis comparé le tableau ainsi reconstitué au tableau initial.

Le tableau MU consiste en des données climatologiques, CI en des données sur les préférences de marques de cigarettes et CA en des teneurs en p.p.m. de sédiments marins.

Pour les tableau CI et CA, plusieurs essais ont été effectués en faisant varier le pourcentage du nombre de valeurs manquantes par rapport au nombre total d'éléments du tableau (card I x card J).

Ces pourcentages varient respectivement de 2 % à 23,3 %, de 0,7 % à 11,6 %.

Il faut noter que les valeurs trouvées pour chaque trou restent pratiquement identiques, quel que soit le nombre total de trous et sont très proches des valeurs réelles en particulier lorsque les valeurs propres sont faibles (par exemple pour le tableau MU).

- Reconstitution d'ordre r

Lors du calcul des nombres K_{rp} le problème des nombres négatifs s'est posé : faut-il les remplacer par zéro soit à chaque tour, soit à la fin. Des nombres négatifs peuvent apparaître lorsqu'on ne considère que peu de valeurs propres et que des composantes des facteurs (F(i) ou G(j)) sont négatives. Ici encore, les résultats sont si voisins que nous ne distinguerons pas ces deux méthodes. Cette recherche d'une reconstitution d'ordre r a été faite pour ces mêmes tableaux, avec $r \leq 5$.

De la même façon, il faut noter que les valeurs trouvées semblent dépendre très peu du nombre de trous, au contraire du nombre d'itérations et donc du temps de calcul.

Dans ces tableaux, la première valeur propre à toujours représenté un important pourcentage de l'inertie du nuage; de même les valeurs suivantes étaient très faibles en valeur absolue. C'est peut-être ce qui explique pourquoi la reconstitution d'ordre 1 a toujours paru suffisante et une bonne reconstitution du tableau original.

Pour comparer les analyses des correspondances

- du tableau complet
- du tableau avec des zéros pour les trous
- du tableau avec les valeurs calculées pour les trous,

on a calculé les coefficients de corrélations entre les valeurs des facteurs pour les éléments de I et de J et les variances.

Pour les premiers axes factoriels considérés dans le cas des tableaux complétés au moyen de l'algorithme TROU, les coefficients de corrélation restent très élevés, de l'ordre de 0,9 et la variance ne dépasse pas 0,2.

Ces mêmes études ont été reprises pour un tableau plus important de mesures de praomys, noté PR pour comparer les résultats obtenus par cet algorithme et ceux obtenus en faisant l'analyse des correspondances après avoir supprimé les colonnes correspondant aux valeurs manquantes. Ici, les coefficients de corrélation ont toujours été très élevés, supérieur à 0,95 et les variances de l'ordre de 0,1. Il faut remarquer que dans ces cas PR comme dans le cas MU ces coefficients plus élevés correspondent à des reconstitutions d'ordre successif très voisines.

5°) Conclusion

En conclusion :

- cet algorithme représente un outil puissant permettant d'utiliser la méthode d'analyse factorielle des correspondances dans le cas de données manquantes ou erronées.

- cet algorithme est d'autant plus sûr que les différentes reconstitutions d'ordre successif 0, 1, ...5 donnent des résultats très proches. Ceci nous donne une valeur indicatrice permettant d'apprécier la validité de cet algorithme.

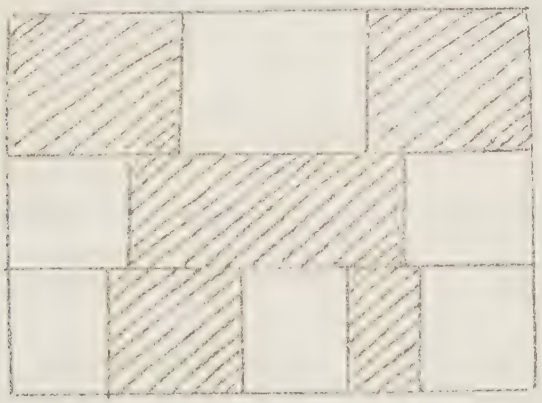
- le nombre de trous ,jusqu'à 23 %, n'intervient que très peu dans les résultats obtenus. Seul le temps de calcul en dépend très directement.

En ce qui concerne les temps de calcul, ce programme a demandé pour rechercher les 6 premières approximations des 7 tableaux (MU, CI9, CI23, CI55, CA11, CA41,CA91) 6 mn d'unité centrale de l'IBM 370-165 du CIRCE.

Cet algorithme TROU permet d'appliquer les méthodes d'analyse factorielle à des tableaux de connées incomplets mais dans lesquels on dispose d'informations suffisamment réparties pour chaque individu ou chaque caractère.

Par contre, deux autre possibilités sont envisageables :

- Dans certains cas, on désire analyser le tableau complet mais ce sont des blocs entiers de la matrice des données qui n'ont pu être collectés ; le tableau des données se présente par exemple selon le schéma suivant:



il n'est pas certain qu'alors les conditions de convergence de l'algorithme TROU soient vérifiées ou que, en raison même de sa construction, cet algorithme donne des résultats de bonne qualité.

C'est en effet toute une partie du phénomène étudié qui a pu disparaître.

Une méthode, étudiée par Monsieur BENSABER, devrait résoudre ce problème. Si nous appelons Q la matrice des covariances à diagonaliser pour effectuer l'analyse du tableau, nous avons

$$q_{jj'} = \sum_{i \in I} \frac{k(i,j)\,k(i,j')}{k(i)\sqrt{k(j)\,k(j')}}$$

Notons I(j) l'ensemble des éléments de I pour lesquels on connaît k(i,j). L'idée est de calculer directement la matrice à diagonaliser en pondérant chaque élément $q_{jj'}$ par card $(I(j) \cap I(j'))$

On obtient :

$$q_{jj'} = \frac{1}{\mathrm{Card}(I(j) \cap I(j'))} \sum_{i \in I} \frac{k(i,j)\,k(i,j')}{k(i)\sqrt{k(j)\,k(j')}}$$

Cette méthode ne reconstitue pas le tableau des données mais devrait donner une bonne approximation de l'analyse factorielle du tableau complet.

- Dans d'autres cas, les résultats de l'analyse factorielle ne sont plus seuls souhaités : on désire également une très bonne approximation du tableau complet des données.

L'algorithme TROU ne résout ce problème que lorsque le tableau initial présente quelques particularités (à l'exemple du MU et PR).

Dans ce cas, l'idée de Monsieur KERBAOL est de comparer les individus colonne par colonne. Lorsque deux individus se trouvent très voisins, on obtiendra la valeur approchée cherchée par comparaison avec celle de l'individu voisin.

II. RAPPEL D'ANALYSE DES CORRESPONDANCES.

La méthode d'analyse des correspondances se rapporte à diverses sortes de tableaux, par exemple

- tableaux de contingence .

I,J s'imposent à priori et les nombres k(i,j) sont des entiers, résultats de comptes.

- tableaux de fréquence .

Sur l'ensemble produit I,J on a une loi de probabilité p(i,j). Le plus souvent les probabilités p(i,j) ne sont estimées que par des fréquences calculées sur un nombre fini d'observations.

- tableaux de mesure .

L'ensemble I est un ensemble d'individus et l'ensemble J un système de mesures choisi de telle sorte que le vecteur $\{k(i,j) \mid j \in J\}$ soit pour l'étude considérée une description satisfaisante de l'individu.

- tableaux de description logique .

Les k(i,j) sont tous égaux à 0 ou à 1, k(i,j) = 1 signifiant que l'individu possède la propriété j, k(i,j) = 0 qu'il ne la possède pas.

- tableaux d'intensité .

k(i,j) est la note (ou le mérite) de l'individu i pour la matière j. Les tableaux précédents sont des cas particuliers de ceux-ci : les notes y sont 0 ou 1; pour les calculs, nous pouvons nous restreindre au cas où nous avons une loi de fréquence f_{IJ} sur l'ensemble $I \times J$.

On s'y ramènera en posant lorsque c'est nécessaire

$$\forall i,j \in I \times J \qquad f_{ij} = \frac{k(i,j)}{k}$$

avec $$k = \sum \{ k(i,j) \mid i \in I, j \in J \}$$

On définit, pour cette loi f_{IJ}, les lois suivantes :

f_J loi marginale sur J $\qquad \forall j \in J \quad f_j = \sum \{ f_{ij} \mid i \in I \}$

f_I loi marginale sur I $\qquad \forall i \in I \quad f_i = \sum \{ f_{ij} \mid j \in J \}$

f_J^i loi conditionnelle de j pour i donné $\qquad \forall j \in J \quad f_j^i = \frac{f_{ij}}{f_i}$

f_I^j loi conditionnelle de i pour j donné $\qquad \forall i \in I \quad f_i^j = \frac{f_{ij}}{f_j}$

f_J^i (resp f_I^j) peut être appelé profil de l'élément i sur J

Le principe de l'analyse des correspondances est de considérer dans R_J, muni de la distance du χ^2 de centre f_J, le nuage N(I) des profils f_J^i affectés respectivement des masses f_i.

Cette distance d est choisie de façon à satisfaire les principes d'équivalence distributionnelle suivants :

- d ne dépend que des probabilités conditionnelles et cumulées sur J
- d ne change pas en cumulant 2 éléments de même profil
- d est d'autant plus petite que les profils des éléments sont semblable

Cette distance du χ^2 $\| f_J^i - f_J^{i'} \|_{f_J}$ est communément appelée distance distributionnelle entre i et i'.

De même, dans R_I, muni de la distance du χ^2 de centre f_I, on considère le nuage N(J) des profils f_I^j affectés respectivement des masses f_j.

Considérons par exemple le nuage N(I) dans R_J. Son centre de gravité est f_J le profil marginal lui-même. Autour de ce centre, le nuage a des axes principaux d'allongement appelés axes factoriels, déterminés comme des axes principaux d'inertie.

Dans R_J, l'axe factoriel de rang α est un vecteur unitaire $u_{\alpha J}$. Notons $\varphi_\alpha^J = u_{\alpha J}/f_J$ la densité de cette mesure.

Alors :

$$\sum \{ \varphi_\alpha^j f_j \mid j \in J \} = 0$$

$$\sum \{ (\varphi_\alpha^j)^2 f_j \mid j \in J \} = 1$$

De plus les axes factoriels successifs sont orthogonaux 2 à 2. D'où :

$$\sum \{ \varphi_\alpha^j \varphi_\beta^j f_j \mid j \in J \} = \delta_{\alpha\beta}$$

où $\delta_{\alpha\beta}$ est le symbole de kronecker ($\delta_{\alpha\beta} = 0$ si $\alpha \neq \beta$ et $\delta_{\alpha\alpha} = 1$

Le nuage N(I) peut alors être rapporté au système, d'origine f_J, de ces axes orthogonaux $u_{\alpha J}$. On note A la suite de ces axes et $F_\alpha(i)$ les coordonnées de chaque f_J^i.

$$f_J^i = f_J + \sum \{ F_\alpha(i) u_{\alpha J} \mid \alpha \in A \}$$

$$f_J^i = f_J \left[1 + \sum \{ F_\alpha(i) \varphi_\alpha^J \mid \alpha \in A \} \right]$$

D'où la formule générale :

$$\forall i,j \in I,J \qquad f_{ij} = f_i f_j \left[1 + \sum \{ F_\alpha(i) \varphi_\alpha^j \mid \alpha \in A \} \right] \qquad (1)$$

Comme l'origine est au centre de gravité f_J, on a

$$\sum \{ F_\alpha(i)\, f_i \mid i \in I \} = 0$$

le moment d'inertie λ_α du nuage dans la direction $u_{\alpha J}$ est par définition :

$$\lambda_\alpha = \sum \{ [F_\alpha(i)]^2 f_i \mid i \in I \}$$

λ_α est nécessairement entre 0 et 1.

Associons à F_α la fonction φ_α^I définie par

$$\varphi_\alpha^i = \lambda_\alpha^{-1/2} F_\alpha(i)$$

On a

$$\sum \{ \varphi_\alpha^i \varphi_\beta^i f_i \mid i \in I \} = \delta_{\alpha\beta}$$

de par la définition des axes principaux d'inertie. Le moment d'inertie totale du système est $\sum \{ \lambda_\alpha \mid \alpha \in A \}$

ou $\sum \{ f_i \| f_j^i - f_j \|^2 f_j \mid i \in I \}$

On note communément τ_α la part de l'inertie totale correspondant à chaque moment λ_α

$$\tau_\alpha = \lambda_\alpha / \sum \{ \lambda_\alpha \mid \alpha \in A \}$$

La formule (1) peut s'écrire sous une forme parfaitement symétrique en i et j

$$f_{ij} = f_i f_j \left[1 + \sum \{ \lambda_\alpha^{1/2} \varphi_\alpha^i \varphi_\alpha^j \mid \alpha \in A \} \right]$$

$$\boxed{f_{ij} = f_i f_j \left[1 + \sum \{ \lambda_\alpha^{-1/2} F_\alpha(i)\, G_\alpha(j) \mid \alpha \in A \} \right]} \qquad (2)$$

avec

$$G_\alpha(j) = \lambda_\alpha^{1/2} \varphi_\alpha^j$$

On a en fait une symétrie totale car, partant du nuage $N(J) \subset R_I$, on aboutit aux mêmes fonctions, aux mêmes valeurs propres.

Cette formule exprimant les f_{ij} par une somme où apparaissent tous les facteurs est exacte.

Si on se borne à ne prendre que les r premiers facteurs (i.e. ceux relatifs aux valeurs propres les plus fortes) on obtient une formule approchée d'ordre r de reconstitution des données en fonction des facteurs :

$$f_{ij} \simeq f_i f_j \left[1 + \sum \{ \lambda_\alpha^{-1/2} F_\alpha(i)\, G_\alpha(j) \mid \alpha = 1, \dots r \} \right]$$

Dans le cas général, nous nous trouvons non pas en présence d'une loi de probabilité f_{IJ} mais d'un tableau de nombres k(i,j) ; nous obtenons la formule :

(I)

$$\boxed{k(i,j) = \frac{k(i)\, k(j)}{k} \left[1 + \sum \{ \lambda_\alpha^{-1/2} F_\alpha(i)\, G_\alpha(j) \mid \alpha \in A \} \right]}$$

avec

$$\forall i \in I \quad k(i) = \sum \{ k(i,j) \mid j \in J \}.$$

$$\forall j \in J \quad k(j) = \sum \{ k(i,j) \mid i \in I \}$$

$$k = \sum \{ k(i,j) \mid i \in I, j \in J \} = \sum \{ k(i) \mid i \in I \} = \sum \{ k(j) \mid j \in J \}$$

III. CONSTRUCTION DES ALGORITHMES.

Dans le rappel précédent, nous sommes arrivés à la formule suivante :

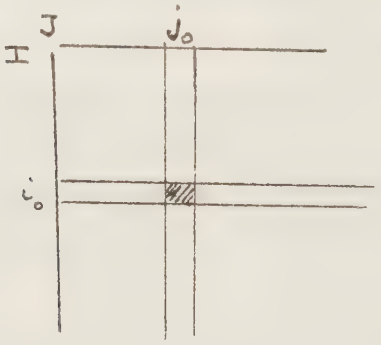
$$(I)\qquad \boxed{\begin{array}{l}\forall i\in I\\ \forall j\in J\end{array}\quad k_{ij} = \frac{k_i k_j}{k}\left[1 + \sum\left\{ \lambda_\alpha^{-1/2} F_\alpha(i) G_\alpha(j) \mid \alpha\in A\right\}\right]}$$

lorsqu'on se limite à l'ensemble A_r des r premières valeurs propres, on obti la meilleure approximation de rang r, au sens des moindres carrés.

Lorsqu'on se trouve en présence de données manquantes, on cherche à remplacer ces "trous" par des nombres, calculés à partir des données connue de façon à obtenir, pour tout couple de point (i,j), la formule (I).

Cas particulier.

Pour préciser, prenons l'exemple d'une seule donnée manquante.



Soit (i_o,j_o) le couple de points de I x J correspondant à la donnée manquante $k_{i_o j_o}$.

Posons $k_{i_o j_o} = 0$ et notons de la manière classique :

$$\forall i\in I \quad k_i = \sum\{k_{ij} \mid j\in J\}$$
$$\forall j\in J \quad k_j = \sum\{k_{ij} \mid i\in I\}$$
$$k = \sum\{k_{ij} \mid i\in I, j\in J\}$$

Notons u la valeur recherchée pour (i_o,j_o).

Nous voulons avoir en notant $K_{ij} = \sum\left\{ \lambda_\alpha^{-1/2} F_\alpha(i) G_\alpha(j) \mid \alpha\in A\right\}$ +

$$\begin{array}{ll}\forall i\in I\ i\neq i_o & (u+k)k_{ij} = k_i k_j K_{ij}\\ \forall j\in J\ j\neq j_o & \end{array}$$

$$\begin{array}{ll}\forall i\in I\ i\neq i_o & (u+k)k_{ij_o} = k_i (k_{j_o}+u) K_{ij_o}\\ \forall j\in J\ j\neq j_o & (u+k)k_{i_o j} = (k_{i_o}+u) k_j K_{i_o j}\\ & (u+k)u = (k_{i_o}+u)(k_{j_o}+u) K_{i_o j_o}\end{array} \qquad (I)$$

Les valeurs K_{ij} sont calculées à partir du tableau des données complété avec u pour le couple (i_o,j_o).

Etant donné la remarque précédente sur les approximations de rang r, un nombre approché u_r d'une approximation de rang r vérifiera les formules :

$$
(I_r) \quad
\begin{cases}
\forall i \in I \quad i \neq i_0 \\
\forall j \in J \quad j \neq j_0 & k_{ij} = \dfrac{k_i k_j}{k + u_r} K_{ij} \\[2ex]
\forall i \in I \quad i \neq i_0 & k_{ij_0} = \dfrac{k_i (k_{j_0} + u_r)}{k + u_r} K_{ij_0} \\[2ex]
\forall j \in J \quad j \neq j_0 & k_{i_0 j} = \dfrac{(k_{i_0} + u_r)\, k_{j_0}}{k + u_r} K_{i_0 j} \\[2ex]
& u_r = \dfrac{(k_{i_0} + u_r)(k_{j_0} + u_r)}{k + u_r} K^r_{i_0 j_0}
\end{cases}
$$

en notant $$K^r_{ij} = \sum \left\{ \lambda_\alpha^{-1/2} F_\alpha(i)\, G_\alpha(j) \mid \alpha \in A_r \right\} + 1$$

Nous appelerons alors u_r une reconstitution d'ordre r.

Pour obtenir un tel nombre u_r , nous pouvons donc procéder de la façon suivante : prendre une reconstitution d'ordre 0, puis d'ordre 1... jusqu'à l'ordre r.

Pour une reconstitútion d'ordre 0 on a

$$A_r = \phi \quad , \quad K^0_{i_0 j_0} = 1$$

Il est à noter que si l'on calcule les nombres K_{ij} à partir du tableau initial avec u = 0, on obtiendra les formules (I) exactes pour $u = k_{i_0 j_0} = 0$.

1°) <u>Recherche des solutions</u> $(k+u)\,u = (k_i + u)(k_j + u)\, K^r_{ij}$

a) <u>Algorithme 1</u>

Supposons-nous donnée une reconstitution d'ordre r - 1. A partir de cette reconstitution, nous pouvons déterminer l'ensemble des nombres K_{ij} et K^r_{ij}.

Soit $\mathcal{N} = \left\{ (i_n, j_n) \mid n \in [1,N] \right\}$ l'ensemble des N "trous" du tableau de données. On recherche un ensemble $\mathcal{U}_r = \left\{ u_n \mid n \in [1,N] \right\}$ de nombres tels que

si $\forall (i,j) \in \mathcal{N} \quad k_{ij} = u_n$, le tableau final vérifie les conditions (I_r).

On va donc partir de k_{ij}^0, k_i^0, k_j^0, k^0, valeurs initiales dans la reconstitution d'ordre r - 1. A partir de là, nous allons définir N suites de nombre $k_{ij}^p \; (i,j) \in \mathcal{N}$

$$\forall (i,j) \in \mathcal{N} \qquad k_{ij}^{1} = \frac{k_i^{0} k_j^{0}}{k^{0}} K_{ij}^{r}$$

$$\forall (i,j) \in I \times J \cdot \quad (i,j) \notin \mathcal{N} \qquad k_{ij}^{1} = k_{ij}^{0}$$

Ce nouveau tableau (k_{ij}^{1}) permet de calculer

k_i^{1}, k_j^{1}, k^{1}. On recommence à partir de ces nouvelles valeurs, et ainsi de suite...

$$\forall (i,j) \in \mathcal{N} \qquad k_{ij}^{p} = \frac{k_i^{p-1} k_j^{p-1}}{k^{p-1}} K_{ij}^{r}$$

$$\forall (i,j) \in I \times J \cdot \quad (i,j) \notin \mathcal{N} \qquad k_{ij}^{p} = k_{ij}^{p-1} = k_{ij}^{0}$$

Le système U_r cherché sera alors les valeurs limites, si elles existent (ce point sera considéré plus loin) des k_{ij}^{p} pour $p \to \infty$, $(i,j) \in \mathcal{N}$

De cette façon, nous avons défini un algorithme noté algorithme 1.

Remarque 1. Une légère variante pourra consister à changer les $k_{ij}^{p}((i,j) \in \mathcal{N})$ non plus par tableau entier à partir du tableau p - 1 mais point par point. i.e. à chaque changement d'un k_{ij}^{p} on fera changer aussi les valeurs de k, k_i, k_j pour le calcul du point suivant. On verra dans les résultats que l'on obtient de cette façon exactement les mêmes résultats.

Remarque 2. On peut aussi pour chaque ordre r repartir du tableau où les valeurs des trous sont toutes ramenées à 0 et avec seul K_{ij}^{r} dépendant du tableau précédent.

b) Algorithme 2.

Dans le cas d'une reconstitution d'ordre 0, nous pouvons faire la remarque suivante :

Nous avons la formule simplifiée :

(II) $$\forall (i,j) \in I \times J \qquad k_{ij} = \frac{k_i k_j}{k}$$

ce qui s'écrit pour un seul point (i_o, j_o) manquant

$$k_{i_o j_o} = \frac{(\bar{k}_i + k_{i_o j_o})(\bar{k}_j + k_{i_o j_o})}{\bar{k} + k_{i_o j_o}}$$

avec

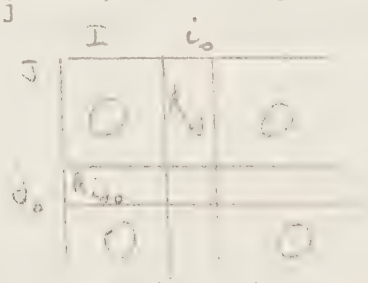
$$\bar{k}_i = \sum \{ k_{ij} \mid j \in J - \{j_o\} \} \qquad \bar{k}_j = \sum \{ k_{ij} \mid i \in I - \{i_o\} \}$$

$$\bar{k} = \sum \{ k_{ij} \mid j \in J - \{j_o\}, i \in I - \{i_o\} \}$$

Cela exige

$$k_{i_o j_o} = \frac{\overline{k_i}\ \overline{k_j}}{k - \overline{k_i} - \overline{k_j}}$$

On suppose $k - \overline{k_i} - \overline{k_j} > 0$, la correspondance serait vraiment trop particulière autrement



Cela nous conduit à penser à l'algorithme suivant pour la reconstitution d'ordre zéro :

on pose :

$$\forall (i,j) \in \mathcal{NP} \qquad k^0_{ij} = 0$$

$$k^0_i = \sum \{ k^0_{ij} \mid j \in J \}$$

$$k^0_j = \sum \{ k^0_{ij} \mid i \in I \}$$

$$k^0 = \sum \{ k^0_{ij} \mid i \in I, j \in J \}$$

alors :

$$k^1_{ij} = \frac{(k^0_i - k^0_{ij})(k^0_j - k^0_{ij})}{k^0 - k^0_i - k^0_j + k^0_{ij}}$$

ou de façon plus générale, avec les mêmes notations

$$k^m_{ij} = \frac{(k^{n-1}_i - k^{n-1}_{ij})(k^{n-1}_j - k^{n-1}_{ij})}{k^{n-1} - k^{n-1}_i - k^{n-1}_j + k^{n-1}_{ij}}$$

De la même façon, le système k_{ij} des nombres cherchés sera le système limite, s'il existe, de cet algorithme appelé n° 2.

Nous verrons plus loin que de cette façon, nous obtenons les mêmes résultats pour un nombre d'itérations moindre.

Il est bien évident que nous obtiendrons une légère variante point par point comme précédemment en calculant après chaque changement de nombre k^n_{ij} les valeurs k, k_i, k_j.

Comme pour l'algorithme 1, nous obtenons les mêmes résultats.

c) Algorithme 3.

Nous allons agir par récurrence sur le nombre de trous. Soit n le nombre de trous.

Nous pouvons considérer ces n trous comme la réunion de n-1 trous ($\in \mathbb{N}$) et d'un dernier soit (i_o, j_o).

Prenant $k^{\circ}_{i_o j_o} = 0$, nous déterminons une solution, pour les (n-1) trous, qui dépend de $k^{\circ}_{i_o j_o}$ puisqu'elle dépend de tout le tableau de nombres (nous supposons l'existence de cette solution comme hypothèse pour notre récurrence).

Le nouveau tableau k'° nous permet alors de calculer une valeur

$$k^{1}_{i_o j_o} = \frac{(k'^{\circ}_{i_o} - k^{\circ}_{i_o j_o})(k'^{\circ}_{j_o} - k^{\circ}_{i_o j_o})}{k'^{\circ} - k'^{\circ}_{i_o} - k'^{\circ}_{j_o} + k^{\circ}_{i_o j_o}}$$

A nouveau, cela détermine une nouvelle solution pour les (n-1) autres trous... et ainsi de suite jusqu'à la convergence avant de passer à n+1 trous.

Comme nous avons une solution pour 1 point par l'algorithme 2, de cette façon nous déterminons une reconstitution de rang r des N points.

d) Algorithme 4.

Partant d'une reconstitution d'ordre r-1 qui détermine pour tout point (i,j) de N des nombres K_{ij} et K^{r}_{ij}, il est bien évident que si l'on utilise pour le premier trou l'algorithme 1 et non plus l'algorithme 2, on obtiendra en opérant de la même façon, un algorithme avec la formule suivante :

$$k^{\wedge}_{i_o j_o} = \frac{k'^{\circ}_{i_o} k'^{\circ}_{j_o}}{k'^{\circ}} K^{r}_{i_o j_o}$$

... etc...

Remarque : Ces 4 algorithmes décrits, il est bien évident que tout autre algorithme de résolution de l'équation F(x) = 0 peut être utilisé pour obtenir ce système de solutions. On pourrait utiliser notamment l'algorithme de Newton-Raphson, mais qui demande à chaque pas la résolution d'un système linéaire. Par la suite, nons appellerons l'un quelconque de ces algorithmes ITER.

2°) Recherche d'une approximation de rang r.

Cet algorithme ITER fournit notamment une approximation de rang 0.

L'approximation de rang r sera obtenue à partir de celle de rang r-1.

On calculera les valeurs propres et vecteurs propres du tableau reconstitué puis par la formule (I) une reconstitution de rang r des valeurs cherchées. On pourra alors à partir de ces nombres utiliser l'algorithme ITER qui nous donnera de nouvelles valeurs pour les données manquantes. Il ne restera plus qu'à recommencer le processus précédent. L'approximation de rang r sera obtenu à la convergence, si elle existe, de cet algorithme.

On peut remarquer qu'on aurait pu ne pas utiliser l'algorithme ITER à chaque pas mais faire varier en même temps k_i, k_j et K_{ij}^{r} pour obtenir les k_{ij} comme limites de suites définies par

$$k_{ij}^{P} = \frac{k_i^{P-1} k_j^{P-1}}{k^{P-1}} K_{ij}^{r,P-1}$$

3°) Meilleure approximation.

Le nombre de valeurs propres étant fini, on obtiendra la meilleure approximation lorsque celle de rang r sera égale à celle de rang r+1, ce qui arrivera au pire pour r = n-1 (n = dim E).

On peut faire la remarque suivante :
Pour la reconstitution d'ordre 0, on a $\forall h \in [1,N]$, $K_h = 1$.

Alors en considérant cette reconstitution, nous avons :

$$\begin{matrix}\forall i \in I \\ \forall j \in J\end{matrix} \qquad k_{ij} = \frac{k_i k_j}{k} \left[1 + \sum \left\{ \lambda_\alpha^{-1/2} F_\alpha(i) G_\alpha(j) \mid \alpha \in A \right\}\right]$$

En particulier pour les trous, on aura :

$$k_{ij} = \frac{k_i k_j}{k}$$

Maintenant, supposons qu'on soit à la reconstitution d'ordre r :

$$k_{ij}^{r} = \frac{k_i^{r} k_j^{r}}{k^{r}} \left[1 + \sum \left\{ \lambda_\alpha^{-1/2} F_\alpha(i) G_\alpha(j) \mid \alpha \in A \right\}\right]$$

A la reconstitution d'ordre r+1, on aura

$$k_{ij}^{r+1} = \frac{k_i^{r+1} k_j^{r+1}}{k^{r+1}} \left[1 + \sum \left\{ \lambda_\alpha^{-1/2} F_\alpha(i) G_\alpha(j) \mid \alpha \in A_{r+1} \right\}\right]$$

Donc losque le terme correctif restant sera négligeable, (à la précision près possible):

$$\sum \left\{ \lambda_\alpha^{-1/2} F_\alpha(i) G_\alpha(j) \mid \alpha \in A, \alpha \notin A_{r+1} \right\}$$

on obtiendra le même système à l'ordre supérieur.

En fait, étant donné la propagation des erreurs d'arrondi et le fait que les valeurs propres sont prises dans l'ordre décroissant, généralement, nous serons amené à ne considérer que les premières approximations.

IV - ETUDE DES ALGORITHMES ITER

1 - Algorithme n° 1

Nous avons vu au chapitre précédent que l'algorithme 1 était déterminé par la formule

$$\forall (i,j) \in \mathcal{K} \qquad k_{ij}^{p} = \frac{k_i^{p-1} k_j^{p-1}}{k^{p-1}} K_{ij}^{r} \qquad (4)$$

où K_{ij}^{r} est une constante fixe pour (i, j) donné.

Pour fixer les idées et pour une meilleure compréhension de l'algorithme 2, nous allons étudier le cas particulier d'une seule donnée manquante et d'une reconstitution d'ordre 0.

a) $\mathcal{K} = \{(i_o, j_o)\}, r = 0$

On a $K_{i_o j_o}^{o} = 1$

La formule (4) entraîne

$$k_{i_o j_o}^{p} = \frac{k_{i_o}^{p-1} k_{j_o}^{p-1}}{k^{p-1}}$$

Partant de $k_{i_o j_o}^{o} = 0$, cela entraîne

$$k_{i_o}^{p-1} = k_{i_o}^{o} + k_{i_o j_o}^{p-1}$$

$$k_{j_o}^{p-1} = k_{j_o}^{o} + k_{i_o j_o}^{p-1}$$

$$k^{p-1} = k^{o} + k_{i_o j_o}^{p-1}$$

D'où

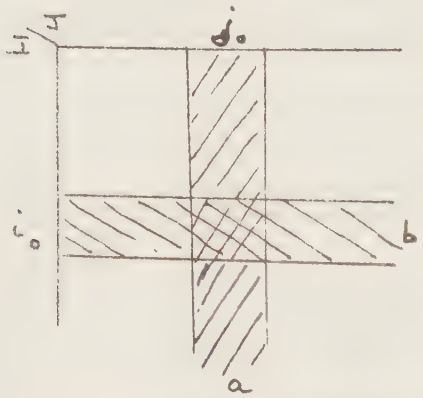
$$k_{i_o j_o}^{p} = \frac{\left(k_{i_o}^{o} + k_{i_o j_o}^{p-1}\right)\left(k_{j_o}^{o} + k_{i_o j_o}^{p-1}\right)}{k^{o} + k_{i_o j_o}^{p-1}}$$

On est ainsi ramené au problème classique suivant : Trouver la limite d'une suite de nombres positifs u_n définie par :

$$u_0 = 0$$

$$u_n = \frac{(a + u_{n-1})(b + u_{n-1})}{c + u_{n-1}}$$

pour lesquels



$$a = k^{\circ}_{i_0} = \sum \{ k_{i_0 j} \mid j \in J - \{j_0\} \}$$

$$b = k^{\circ}_{j_0} = \sum \{ k_{i j_0} \mid i \in I - \{i_0\} \}$$

$$c = k^{\circ} = \sum \{ k_{ij} \mid (i,j) \in I \times J - \{(i_0, j_0)\} \}$$

nous supposons $c > a+b$ pour avoir une correspondance réelle entre I et J et non deux correspondances séparées

$$i_0 \times \{ J - \{j_0\} \}$$

$$J_0 \times \{ I - \{i_0\} \}$$

Pour cela, étudions la fonction

$$Y = \frac{(a + x)\ (b + x)}{c + x}$$

$$y' = \frac{x^2 + 2cx + c(a + b) - ab}{(c + x)^2}$$

$$y'' = \frac{2}{(C + x)^3}\ (c^2 - c(a + b) + ab)$$

or $c > a + b \Rightarrow$ $y'' > 0$

$y' > 0$ pour $x > 0$

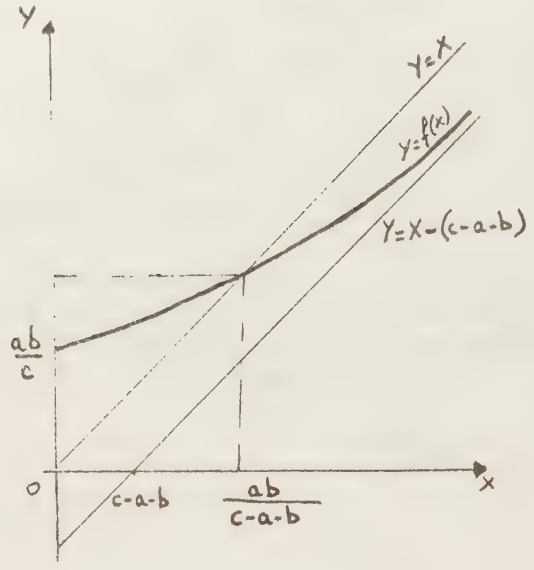
Si $x \to +\infty$, $\frac{y}{x} \to 1$

$$y - x = \frac{(a + b - c)\ x + ab}{c + x}$$

Donc lorsque $x \to +\infty$, y a pour asymptote la droite

$$Y = X + (a + b - c)$$

Nous obtenons la courbe y = f(x) suivante



Partant de $u_o = o$, on obtient $u_1 = \frac{ab}{c}$ .....

Puisque $y' > o$, il est clair que $u_n > u_{n-1}$

De plus, on a

$$u_1 \leq \frac{ab}{c-(a+b)}$$

Or

$$f\left(\frac{ab}{c-(a+b)}\right) = \frac{ab}{c-(a+b)}$$

D'où $\forall u$, on obtient

$$u_{n-1} < u_n \leq \frac{ab}{c-(a+b)}$$

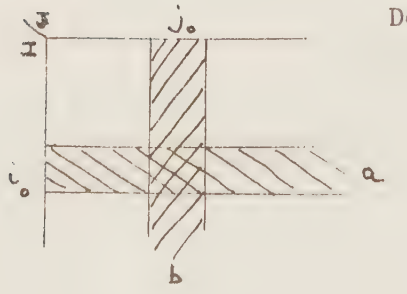
On est donc en présence d'une suite bornée de nombres croissants positifs. Cette suite est donc convergente et sa limite est

$$\frac{ab}{c-(a+b)} .$$

Il est à remarquer que si l'on part de $u_o > \frac{ab}{c-(a+b)}$, on obtient alors une suite décroissante tendant vers cette même limite $\frac{ab}{c-(a+b)}$.

C'est cette solution particulière qui nous a permis de considérer l'algorithme n°2.

b) $\mathcal{N} = \{(i_o, j_o)\}$, r quelconque



De même, nous retrouverons la suite de nombres définis pa

$$u_o = o$$

$$u_n = \frac{(a+u_{n-1})(b+u_{n-1})}{c+u_{n-1}} \cdot K = f(u_{n-1})$$

où $K = K^r_{i_o j_o}$

$$a = k^=_{i_o} = \Sigma\{k_{i_o j} \mid j \in J - \{j_o\}\}$$

$$b = k^\circ_{j_o} = \Sigma\{k_{i j_o} \mid i \in I - \{i_o\}\}$$

$$c = k^\circ = \Sigma\{k_{ij} \mid (i,j) \in I \times J - \{(i_o, j_o)\}\}$$

Pour les mêmes raisons qu'en a, nous supposerons $c > a+b$.

Le cas $K \leq 1$ se déduit immédiatement du cas précédent ($K = 1$). On obtient alors une limite unique stable qui est la solution positive $\mathcal{U}$ de l'équation

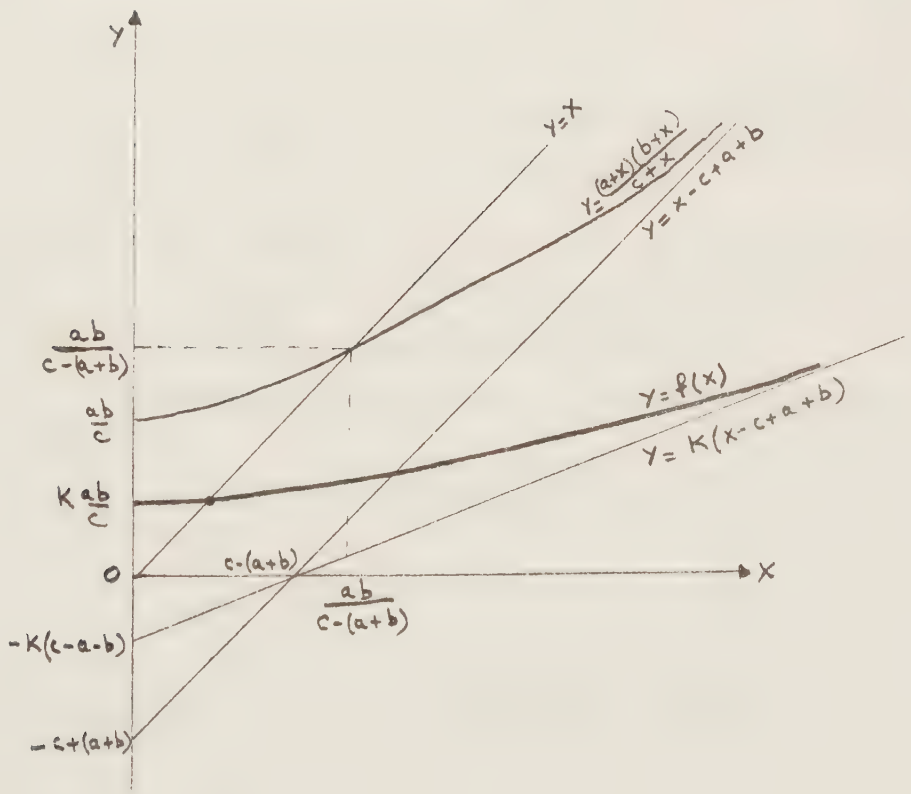
$$x = K \frac{(a + x)(b + x)}{c + x}$$

Cette limite ne dépend pas du choix de u_0 $(u_0 \geq 0)$

Si $u_0 > \mathcal{U}$, on a une suite décroissante

$u_0 < \mathcal{U}$, on a une suite croissante

$u_0 = \mathcal{U}$, on a $u_n = \mathcal{U}$ $\forall n$



Cas $K \geq 1$

Pour avoir une limite à la suite u_n définie par (5), nous devons avoir une solution positive à l'équation

$$x = K \frac{(a + x)(b + x)}{c + x} \qquad (6)$$

Il est bien évident que si nous nous trouvons dans le cas de la formule exacte (I) et non pas d'une approximation, K est calculé de telle façon que l'équation (6) est vérifiée par le $k_{i_o j_o}$ qui a permis de déterminer $K_{i_o j_o}$. C'est un résultat classique de la méthode d'analyse des correspondances. Mais, nous allons étudier de façon plus générale l'équation (6) pour une approximation de rang r.

L'équation (6) est équivalente à :

$$X^2(K-1) - (c - K(a+b)\, x + Kab = 0 \qquad (7)$$

Cette équation admet des solutions si et seulement si $\Delta(K) \geq 0$

$$\Delta(K) = (c - K(a + b))^2 + 4\, Kab(1-K)$$
$$= K^2 (a-b)^2 - 2K (c(a+b)-2ab) + c^2 \qquad (8)$$

• $a \neq b$; l'équation $\Delta(K) = 0$ a un discriminant δ toujours positif :

$$\delta = [c(a+b) - 2ab]^2 - c^2(a-b)^2$$
$$= 4ab\,[c\,(c-(a+b)) + ab] = 4ab\,(c-a)(c-b) > 0$$

Si K_1 et K_2 sont les racines de l'équation $\Delta(K)=0$, K_1 et K_2 sont déterminés par

$$K_1+K_2 = \frac{2(c(a+b)-2ab)}{(a-b)^2} \quad ; \quad K_1 + K_2 > 0$$

$$K_1 K_2 = \frac{c^2}{(a-b)^2} \quad ; \quad K_1 K_2 > 0$$

$$K_1 = \frac{c^2}{\left[\sqrt{a(c-b)} + \sqrt{b(c-a)}\right]^2} \qquad K_2 = \frac{c^2}{\left[\sqrt{a(c-b)} - \sqrt{b(c-a)}\right]^2}$$

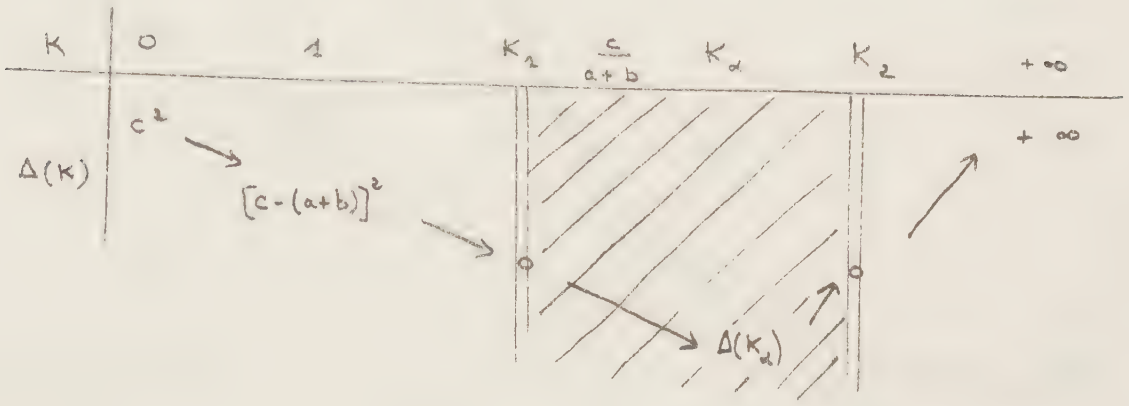
De plus

$$\Delta(1) = (a-b)^2 + 2\,(c(a+b) - 2ab) + c^2$$
$$= [c - (a+b)]^2$$

$$\Delta'(K) = 2\left[(a-b)^2 K - (c(a+b) - 2ab)\right]$$

Donc Δ' est nul pour $K_\alpha = \dfrac{c(a+b) - 2ab}{(a-b)^2}$

D'où la variation de $\Delta(K)$:



De plus, pour que (7) admette une solution positive, le produit des racines $\dfrac{Kab}{K-1}$ étant positif, on doit avoir la somme des racines positive soit

$$\frac{c - K(a+b)}{K-1} > 0$$

c'est à dire

$$K < \frac{c}{a+b}$$

Or

$$\Delta\left(\frac{c}{a+b}\right) = \frac{c^2(a-b)^2}{(a+b)^2} - \frac{2c}{a+b}\left[c(a+b) - 2ab\right] + c^2$$

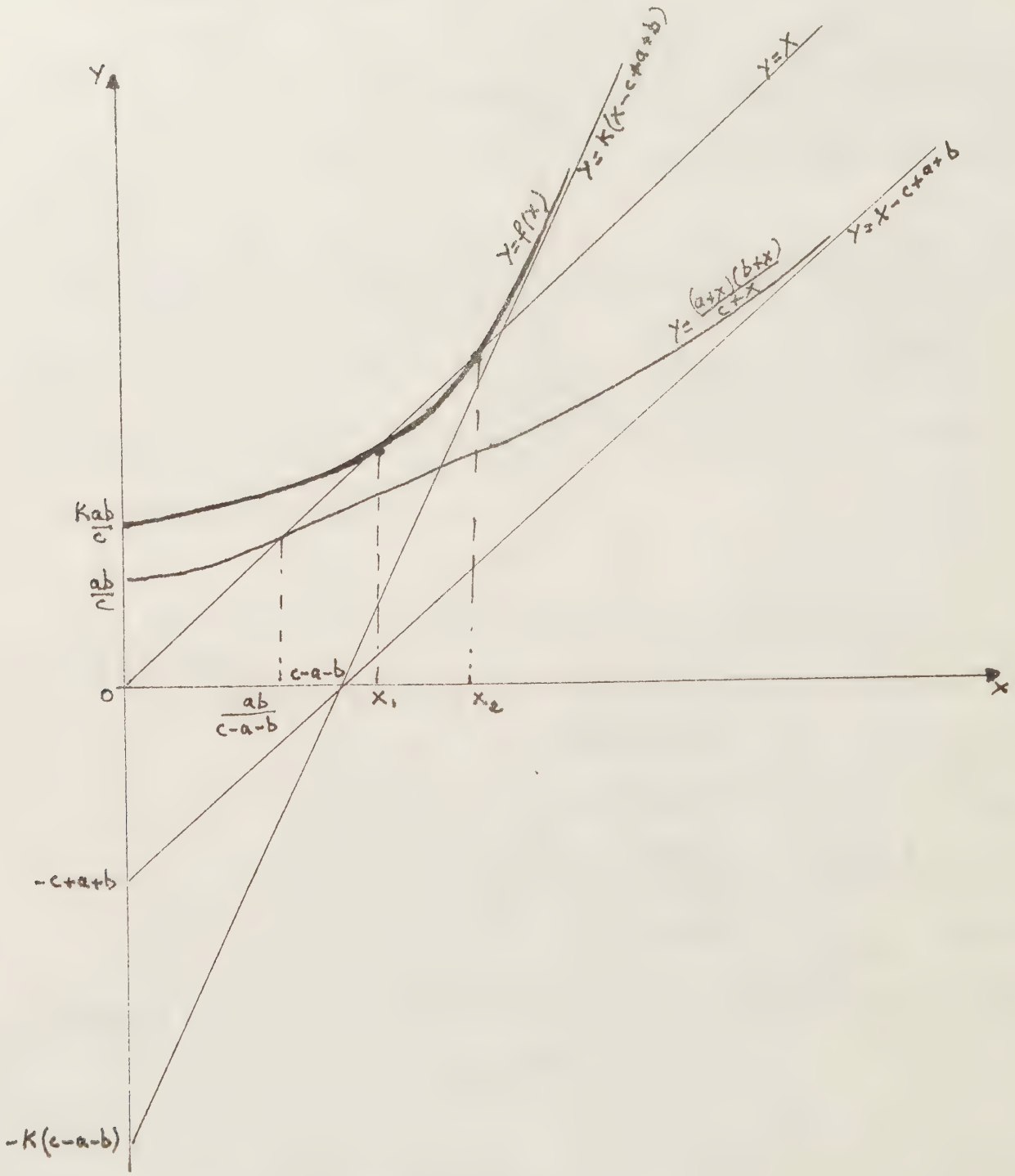
$$= \frac{4cab}{(a+b)^2}\left[-c + a + b\right]$$

Donc $\Delta\left(\dfrac{c}{a+b}\right) < 0$

La seule possibilité pour avoir une solution limite positive à la suite u_n définie par (5) est d'avoir

$$K \leq K_1$$

La représentation graphique de $y = f(x)$ est alors pour $1 < K < K_1$



Sur ce graphique, il est évident que

pour $0 \leq u_0 < x_1$ la suite converge vers x_1 $(u_n \to x_1)$

$u_0 = x_1$ la suite est constante $u_n = x_1$

$x_1 < u_0 < x_2$ la suite converge vers x_1 $(u_n \to x_1)$

$u_0 = x_2$ la suite est constante $u_n = x_2$

$u_0 > x_2$ la suite est divergente $(u_n \to +\infty)$

si $K = K_1$, X_1 et X_2 sont confondus en X_2.

Dans ce cas,

pour $0 \leq u_0 < X_2$ la suite converge vers X_2

$u_0 = X_2$ la suite est constante

$u_0 > X_2$ la suite diverge

• <u>a = b</u> ; L'équation (7) s'écrit

$$X^2(K-1) - (c - 2Ka)X + Ka^2 = 0$$

$$\Delta(K) = c^2 - (c-a)4Ka$$

$$\Delta(K) \geq 0 \iff K \leq \frac{c^2}{4a(c-a)}$$

Il existe donc une solution positive si

$$K \leq \frac{c^2}{4a(c-a)}$$

et

$$K < \frac{c}{2a}$$

On peut remarquer que ceci n'est qu'un cas particulier des conditions précédentes ($K \leq K_1$, $K < \frac{c}{a+b}$) pour lesquelles a = b.

Revenons au cas général, pour avoir une solution positive de (6) nous devons avoir :

$$x(c+x) = K(a+x)(b+x)$$

soit

$$K = \frac{x(c+x)}{(a+x)(b+x)}$$

Etudions cette courbe $K = g(x)$

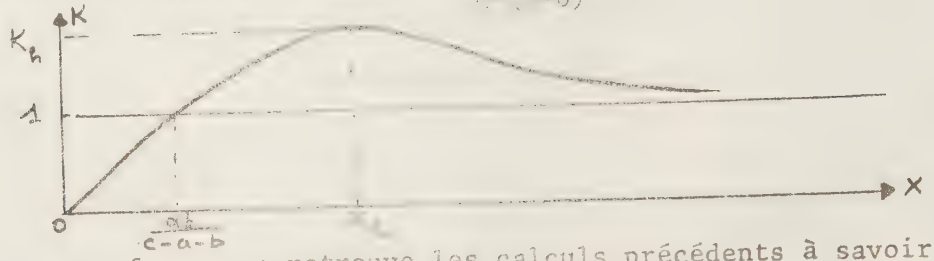
$x \to +\infty \qquad g(x) \to 1$

$$g'(x) = \frac{1}{(a+x)^2(b+x)^2}\left[(2x+c)(a+x)(b+x) - x(c+x)(2x+a+b)\right]$$

$$= \frac{1}{(a+x)^2(b+x)^2}\left[-x^2(c-a-b) + 2abx + cab\right]$$

x	0	$\frac{ab}{c-a-b}$	X_R	$+\infty$
$g(x)$	0 ↗	1	K_R ↘	1

D'après a), $g(x) = 1 + \dots$



De cette façon, on retrouve les calculs précédents à savoir

$$K_h = K_1 \qquad \text{et} \qquad X_h = X_1$$

X_1 est donc la racine positive de

$$x^2(c - a - b) - 2abx - cab = 0$$

K_1 la plus petite racine de

$$K^2(a - b)^2 - 2K\,(c(a + b) - 2ab) + c^2 = 0$$

K_1 et X_1 sont alors liés par la relation (6)

Dans le cas d'une approximation d'ordre r, il ne pourra donc y avoir convergence que si le nombre calculé $K = K_{i_r}$ vérifie

$$\text{II} \quad \begin{cases} K < \dfrac{c}{a+b} \\ \text{et } \Delta(K) \geq 0 \end{cases} \qquad \text{avec} \qquad \Delta(K) = [c - K(a+b)]^2 - 4K(K-1)ab$$

soit $K \leq \dfrac{c^2}{\left[\sqrt{a(c-b)} + \sqrt{b(c-a)}\right]^2}$ la première inégalité se trouvant alors forcément vérifiée.

De plus, il faut remarquer, que cette convergence vers X_1 n'a lieu que si le point u_o de départ de l'algorithme est tel que

$u_o = X_2$

Mais si la convergence a eu lieu à l'ordre r-1, nous avons vu que l'algorithme converge toujours, s'il converge, vers la plus petite valeur X_1.

D'où le schéma suivant qui nous montre que cette dernière condition $(u_0 \leq X_2)$ se trouve toujours vérifiée :

A l'ordre r-1, on a vu la convergence vers X_1.

Donc, en prenant les approximations K' ou K'' vérifiant le système II on a toujours $u_\infty = X_1$ tel que

$$u_0 = X'_2$$

$$u_0 \neq X''_2$$

Si l'on préfère on peut commencer l'algorithme à partir de $u_o = 0$ mais on obtient forcément le même résultat.

c-$\mathcal{S}$ quelconque, r quelconque

Supposons que les N trous se trouvent tous situés dans des colonnes ou des lignes distinctes $(i.e. \quad \forall h,k \in [1,N] \quad h \neq k \Rightarrow i_h \neq i_k \, , j_h \neq j_k)$

Pour simplifier l'écriture, notons de façon similaire

$$\forall h \in [1,N] \qquad K_h = K''_{i_h j_h}$$

$$a_h = \sum \{ k_{i_h j} \mid j \in J - \{j_h\} \}$$

$$b_h = \sum \{ k_{i j_h} \mid i \in I - \{i_h\} \}$$

$$c = \sum \{ k_{ij} \mid (i,j) \in I \times J - \mathcal{N} \}$$

Les N suites u_h seront alors définies par

$$\begin{cases} u_{h,0} = 0 \\ u_{h,n} = K_h \dfrac{(a_h + u_{h,n-1})(b_h + u_{h,n-1})}{c + u_{h,n-1} + \sum\limits_{\ell \neq h} u_{\ell,n-1}} \end{cases}$$

Il faut remarquer les inégalités suivantes :

$$u_{h,n} < K_h \frac{(a_h + u_{h,n-1})(b_h + u_{h,n-1})}{c + u_{h,n-1}} = f(u_{h,n-1})$$

Cette fonction f est une fonction croissante de u pour u positif.

On définit ainsi, une suite majorante v_h, celle qu'on aurait obtenue si le point h avait été le seul trou, i.e celle définie par

$$v_{h,0} = 0$$

$$v_{h,n} = K_h \frac{(a_h + v_{h,n-1})(b_h + v_{h,n-1})}{c + v_{h,n-1}}$$

d'après b), cette suite est convergente si et seulement si :

$$K_h \leq \frac{c^2}{\left[\sqrt{a_h(c-b_h)} + \sqrt{b_h(c-a_h)}\right]^2}$$

Supposons cette condition vérifiée pour tout h, nous obtenons alors :

$$\forall h \in [1,N] \qquad \forall n \geq 0 \qquad u_{h,n} < v_h$$

en notant v_h la limite de la suite $v_{h,n}$

Etudions la fonction $f(x,y) = K \frac{(a+x)(b+x)}{c+y+x}$

$$f'_x = K\left[\frac{a+b+2x}{c+x+y} - \frac{(a+x)(b+x)}{(c+x+y)^2}\right] = K \frac{x^2 + 2(c+y)x + (a+b)(c+y) - ab}{(c+x+y)^2}$$

$$f'_y = -K \frac{(a+x)(b+x)}{(c+x+y)^2}$$

$$f''_{x^2} = K\left[\frac{2(c+y)(c+y-a-b+ab)}{(c+x+y)^2}\right]$$

$$f''_{y^2} = K \frac{(a+x)(b+x)}{(c+x+y)^3}$$

$$f''_{xy} = -K\left[\frac{2(c+y-a-b)x + (a+b)(c+y) - 2ab}{(c+x+y)^3}\right]$$

Si x et y sont positifs, on obtient pour $x \leq X_0$

$$|f(x_1,y_1) - f(x_2,y_2)| \leq K|x_1-x_2|\left[\frac{a+b+2X_0}{c+X_0} - \frac{(a+X_0)(b+X_0)}{(c+X_0)^2}\right] + K|y_1-y_2|\,\frac{(a+X_0)(b+X_0)}{(c+X_0)^2}$$

Appliquant ceci aux suites $u_{h,n}$:

$$|u_{h,n+1} - u_{h,n}| \leq K_h \max_{h\in[1,N]}\left(|u_{h,n}-u_{h,n-1}|\right)\left[\frac{a_h+b_h-2\sqrt{v_h}}{c+\sqrt{v_h}} - \frac{(a_h+\sqrt{v_h})(b_h+\sqrt{v_h})}{(c+\sqrt{v_h})^2} + (N-1)\frac{(a_h+\sqrt{v_h})(b_h+\sqrt{v_h})}{(c+\sqrt{v_h})^2}\right]$$

or $$v_h = K_h\,\frac{(a_h+\sqrt{v_h})(b_h+\sqrt{v_h})}{c+\sqrt{v_h}}$$

$$\Rightarrow \quad |u_{h,n+1} - u_{h,n}| \leq \max_{h\in[1,N]}\left(|u_{h,n}-u_{h,n-1}|\right) \times \frac{K_h(a_h+b_h)+2\sqrt{v_h}K_h+(N-2)\sqrt{v_h}}{c+\sqrt{v_h}}$$

or

$$\frac{K_h(a_h+b_h)+2\sqrt{v_h}K_h+(N-2)\sqrt{v_h}}{c+\sqrt{v_h}} < 1$$

$$\Longleftrightarrow \quad (i)\begin{cases} v_h < \dfrac{c-K_h(a_h+b_h)}{2K_h+N-3} & \text{si} \quad N \geq 3 \\ N=2 \quad K \leq \frac{1}{2} & \text{toujours vérifié .} \\ K > \frac{1}{2} & v_h < \dfrac{c-K_h(a_h+b_h)}{2K_h-1} \end{cases}$$

Soit $P(X) = (K_h-1)X^2 - (c-K_h(a_h+b_h))X + K_h a_h b_h$

Alors V_h est telle que $P(V_h)=0$ et si $K_h>1$ $\quad v_h < \dfrac{c-K_h(a_h+b_h)}{2(K_h-1)}$

Les conditions (i) sont alors équivalentes à

$$P\left(\frac{c-K_h(a_h+b_h)}{2K_h+N-3}\right) < 0 \qquad \text{pour} \qquad \begin{matrix} N \geq 3 \\ N=2 \quad K>\frac{1}{2} \end{matrix}$$

Or $$P\left(\frac{c-K_h(a_h+b_h)}{2K_h+N-3}\right) = \frac{(K_h-1)\left[c-K_h(a_h+b_h)\right]^2}{(2K_h+N-3)^2} - \frac{\left[c-K_h(a_h+b_h)\right]^2}{2K_h+N-3} + K_h a_h b_h$$

$$= \frac{\left[c-K_h(a_h+b_h)\right]^2}{\left[2K_h+(N-3)\right]^2}\left[K_h-1-2K_h-N+3\right] + K_h a_h b_h$$

On doit donc avoir

$$\left[c-K_h(a_h+b_h)\right]^2 > K_h\,\frac{(2K_h+N-3)^2}{K_h+N-2}\,a_h b_h$$

Or on peut remarquer que si N = 2, $K_h \leq \frac{1}{2}$ on a

$$\left[c - K_h(a_h+b_h)\right]^2 > \left[c - \tfrac{1}{2}(a_h+b_h)\right]^2 > a_h b_h > (2K_h-1)^2 a_h b_h$$

car $\quad c^2 - c(a_h+b_h) + \frac{1}{4}(a_h^2 + 2a_h b_h + b_h^2) - a_h b_h = c\left(c-(a_h+b_h)\right) + \frac{1}{4}(a_h-b_h)^2 > 0$

Donc si $\quad \forall h \in [1,N] \quad \left[c - K_h(a_h+b_h)\right]^2 > K_h \dfrac{(2K_h+N-3)^2}{K_h+N-2} a_h b_h$

on a $\quad |u_{h,n+1} - u_{h,n}| \leq \max\limits_{h\in[1,N]} (|u_{h,n} - u_{h,n-1}|) \times \alpha_h$

avec $\quad \alpha_h < 1$

Posant $\quad \alpha = \max\limits_{h\in[1,N]} (\alpha_h) \quad , \quad \alpha < 1$

$$\forall h \in [1,N] \quad |u_{h,n+1} - u_{h,n}| \leq \alpha \max_{h\in[1,N]} (|u_{h,n} - u_{h,n-1}|)$$

Les suites $u_{k,n}$ seront convergentes dans ce cas .

De plus

$$\frac{K_h(2K_h+N-3)^2}{K_h+N-2} - 4K_h(K_h-1) = \frac{K_h(N-1)^2}{K_h+N-2} > 0$$

Donc les conditions précédentes entraînent bien aussi l'existence des V_k.

L'algorithme sera convergent si

$$\forall h \in [1,N] \qquad \left[c - K_h(a_h+b_h)\right]^2 > K_h a_h b_h \frac{(2K_h+N-3)^2}{K_h+N-2}$$

- Supposons que les N trous se trouvent situés dans une seule et même colonne ou ligne (i.e. $\forall h \in [1,N] \quad i_h = i_o$ par exemple)

Notons de même

$$K_h = K^*_{i_o j_h}$$

$$a = \sum \{ k_{i_o j} \mid j \in J - \{j_h \mid h \in [1,N]\} \}$$

$$b_h = \sum \{ k_{i j_h} \mid i \in I - \{i_o\} \}$$

$$c = \sum \{ k_{ij} \mid (i,j) \in I \times J - \mathcal{N} \}$$

On obtient les suites $u_{h,n}$ définies par :

$$u_{h,0} = 0$$

$$u_{h,n} = K_h \frac{(b_h + u_{h,n-1})\left(a + \sum\limits_{k=1}^{N} u_{k,n-1}\right)}{c + \sum\limits_{k=1}^{N} u_{k,n-1}}$$

ce qui entraîne

$$u_{h,n} \geq K_h \frac{(a + u_{h,n-1})(b_h + u_{h,n-1})}{c + u_{h,n-1}} = f(u_{h,n-1})$$

f est une fonction croissante de $u_{h,n-1}$

Considérons les suites $V_{h,n}$ définies par

$$V_{h,0} = 0$$

$$V_{h,n} = K_h \frac{(a + V_{h,n-1})(b_h + V_{h,n-1})}{c + V_{h,n-1}}$$

Si $V_{h,n-1} \leq u_{h,n-1}$

$$u_{h,n} \geq f(u_{h,n-1}) > f(V_{h,n-1}) = V_{h,n}$$

Cette condition est vérifiée pour 0 puisqu'on a

$$u_{h,0} = V_{h,0} = 0$$

On obtient ainsi des suites minorantes. Ces suites sont divergentes lorsque la condition (II) n'est pas vérifiée.

Cela nous donne les conditions nécessaires suivantes

$$\forall h \in [1,N] \qquad K_h \leq \frac{c^2}{\left[\sqrt{(c-b_h)a} + \sqrt{b_h(c-a)}\right]^2}$$

Posons $\alpha = \sup\{K_h \mid h \in [1,N]\}$, $\beta = \inf\{K_h \mid h \in [1,N]\}$

$$u_{h,n} = K_h \frac{(b_h + u_{h,n-1})(a + \sum_{k=1}^{N} u_{k,n-1})}{c + \sum_{k=1}^{N} u_{k,n-1}}$$

$$\sum_{h=1}^{N} u_{h,n} = \frac{(a + \sum_{k=1}^{N} u_{k,n-1})}{(c + \sum_{k=1}^{N} u_{k,n-1})} \left(\sum_{k=1}^{N} K_k b_k + \sum_{k=1}^{N} K_k u_{k,n-1} \right)$$

$$\beta \frac{(a + \sum_{k=1}^{N} u_{k,n-1})}{(c + \sum_{k=1}^{N} u_{k,n-1})} \left(\frac{1}{\beta} \sum_{k=1}^{N} K_k b_k + \sum_{k=1}^{N} u_{k,n-1} \right) \leq \sum_{h=1}^{N} u_{h,n} \leq \alpha \frac{(a + \sum_{k=1}^{N} u_{k,n-1})}{(c + \sum_{k=1}^{N} u_{k,n-1})} \left(\frac{1}{\alpha} \sum_{k=1}^{N} K_k b_k + \sum_{k=1}^{N} u_{k,n-1} \right)$$

Considérons la suite v_n définie par

$$v_0 = 0$$

$$v_n = \alpha \frac{(a + v_{n-1})\left(\frac{1}{\alpha} \sum_{k=1}^{N} K_k b_k + v_{n-1}\right)}{c + v_{n-1}} = g(v_{n-1})$$

g est une fonction croissante de v

$u_{h,0} \leq V_0$

la condition $\sum u_{h,n-1} \leq V_{n-1}$ entraîne

$$\sum u_{h,n} \leq g\left(\sum u_{h,n-1}\right) \leq g(V_{n-1}) = V_n$$

Par recurrence, on obtient $\sum u_{h,n} \leq V_n$

La suite des V_n est convergente si et seulement si

$$\begin{cases} a\alpha + \sum_{h=1}^{N} K_h b_h \leq c \\ \left[c - \left(a\alpha + \sum_{h=1}^{N} K_h b_h\right)\right]^2 - 4(\alpha - 1)\, a \sum_{h=1}^{N} K_h b_h \geq 0 \end{cases}$$

ou

$$\sqrt{a\left(\sup_{h\in[1,N]} K_h \cdot c - \sum_{h=1}^{N} K_h b_h\right)} - 4\left(\sup_{h\in[1,N]} K_h - 1\right) a \left(\sum_{h=1}^{N} K_h b_h\right) \geq 0$$

Or il revient au même de dire que ces conditions se trouvent vérifiées pour

$\sup_{h\in[1,N]} K_h$ ou tout K_h

soit

$$\forall h \in [1,N] \quad \begin{cases} a K_h + \sum_{h=1}^{N} K_h b_h \leq c \\ \left[c - \left(a K_h + \sum_{h=1}^{N} K_h b_h\right)\right]^2 - 4 (K_h - 1) a \sum_{h=1}^{N} K_h b_h \geq 0 \end{cases}$$

ou

$$\forall h \in [1,N] \quad \sqrt{a\left(K_h c - \sum_{h=1}^{N} K_h b_h\right)} - 4 (K_h - 1) a \sum_{h=1}^{N} K_h b_h \geq 0$$

On obtiendra de même une condition nécessaire de convergence en considérant β et non plus α, i.e

$$\sqrt{a\left(\inf_{h\in[1,N]} K_h c - \sum_{h=1}^{N} K_h b_h\right)} + \sqrt{\sum_{h=1}^{N} K_h b_h (c-a)} \leq c$$

Considérons la fonction $f(x,y) = K \dfrac{(b+x)(a+y)}{c+y}$

$$df = K \frac{a+y}{c+y} dx + K \frac{(b+x)(c-a)}{(c+y)^2} dy$$

On a toujours $c > a$.

Si x et y sont positifs et croissants, f(x,y) croit.

Si tous les $u_{h,n}$ croissent, $\sum_{h=1}^{N} u_{h,n}$ croit aussi et donc

$$u_{h,n+1} = f\left(u_{h,n}, \sum_{h=1}^{N} u_{h,n}\right) \geq u_{h,n}$$

Il en résulte que les suites $u_{k,n}$ sont toutes croissantes.

Elles sont donc convergentes si : $\forall h \in [1,N]$

$$\sqrt{a\left(K_h c - \sum_{h=1}^{N} K_h b_h\right)} + \sqrt{\sum_{h=1}^{N} K_h b_h (c-a)} \leq c$$

~~Supposons~~

utilisant les notations précédentes, nous arrivons à des suites de la forme

$$u_{h,0} = 0$$

$$u_{h,n} = K_h \frac{(a_h + x_{h,n-1} + y_{h,n-1})(b_h + x_{h,n-1} + z_{h,n-1})}{c + x_{h,n-1} + y_{h,n-1} + z_{h,n-1} + t_{h,n-1}}$$

où $y_{h,n}$ représente la somme des valeurs des "trous" se trouvant dans la même ligne que celui représenté par $x_{h,n}$, $z_{h,n}$ de celles des "trous" se trouvant dans la même colonne et $t_{h,n}$ de celles des "trous" se trouvant ni dans la même ligne ni dans la même colonne.

Posons

$$a = \sup \{ a_h \mid h \in [1,N] \}$$

$$b = \sup \{ b_h \mid h \in [1,N] \}$$

$$V_n = y_{h,n} + x_{h,n} + z_{h,n} + t_{h,n}$$

$$\Rightarrow \quad x_{h,n} \leq K_h \frac{(a + \sqrt{V_{n-1}})(b + \sqrt{V_{n-1}})}{c + V_{n-1}}$$

$$V_n \leq \left(\sum_{h=1}^{n} K_h\right) \frac{(a + \sqrt{V_{n-1}})(b + \sqrt{V_{n-1}})}{c + V_{n-1}} = f(V_{n-1})$$

f est une fonction croissante de V $(V > 0)$

Considérons la suite définie par $W_0 = 0$

$$W_n = f(W_{n-1})$$

Cette suite est convergente si et seulement si

$$\begin{cases} \sum_{h=1}^{N} K_h < \dfrac{c}{a+b} \\ \left[c - \left(\sum_{h=1}^{N} K_h\right)(a+b)\right]^2 - 4\left(\sum_{h=1}^{N} K_h\right)\left(\sum_{h=1}^{N} K_h - 1\right)ab \geq 0 \end{cases}$$

ce qui est équivalent à

$$\sum_{k=1}^{N} K_k \leq \frac{c^2}{\left[\sqrt{a(c-b)} + \sqrt{b(c-a)}\right]^2}$$

Dans $\mathbb{R}^N$, étudions une suite de la forme

$$X_n = F(X_{n-1})$$

i.e définie par

$$X_{1,n} = F_1(X_{1,n-1} \cdots X_{N,n-1})$$

$$X_{N,n} = F_N(X_{1,n-1} \cdots X_{N,n-1})$$

on sait que cette suite est convergente si la fonction F vérifie une condition de Lipschitz $\|F(X_n) - F(X_{n-1})\| \leq A \|X_n - X_{n-1}\|$ avec $A < 1$
ce qui se trouve réalisé notamment si la norme du Jacobien est inférieure à 1. En particulier, on obtiendra une condition suffisante de convergence :

$$\forall k \in [1,N] \qquad \left|\frac{\partial F_k}{\partial X_1}\right| + \cdots + \left|\frac{\partial F_k}{\partial X_N}\right| < 1$$

Considérons la fonction G : $G = K \dfrac{(a+x+y)(b+x+z)}{c+x+y+z+t}$

$$\Delta = \left|\frac{\partial G}{\partial x}\right| + \left|\frac{\partial G}{\partial y}\right| + \left|\frac{\partial G}{\partial z}\right| + \left|\frac{\partial G}{\partial t}\right| = \frac{\partial G}{\partial x} + \frac{\partial G}{\partial y} + \frac{\partial G}{\partial z} - \frac{\partial G}{\partial t}$$

Notons N_y le nombre de "trous" dans la même ligne qu'un certain "trou" (i,j),

N_z le nombre de "trous" dans la même colonne

N_t le nombre de "trous" ni dans la même colonne, ni dans la même ligne.

On a $N_y + N_z + N_t + 1 = N$

et lorsqu'on calcule la valeur de ce trou en utilisant l'algorithme 1, on utiliser la fonction G dans laquelle x représente la valeur cherchée, y la somme de N_y variables, z la somme de N_z variables, t la somme de N_t variables.

D'où une expression de la forme

$$\Delta = \frac{\partial G}{\partial x} + N_y \frac{\partial G}{\partial y} + N_z \frac{\partial G}{\partial z} - N_t \frac{\partial G}{\partial t}$$

$$\frac{\Delta}{K} = (1+N_y)\frac{b+x+z}{c+x+y+z+t} + (1+N_z)\frac{a+x+y}{c+x+y+z+t} + (2N_t - N)\frac{(a+x+y)(b+x+z)}{(c+x+y+z+t)^2}$$

Si on connaît un intervalle de variation pour x, y, z, t (ce qui est le cas lorsque la suite majorante est convergente puisque toutes les valeurs sont positives ou nulles), on obtiendra de cette façon un système de N conditions, suffisant pour la convergente de l'algorithme 1.

Pour chaque point (i,j) de N, supposons qu'on ait

$$R(i,j) = x \leq x_0$$

$$\Sigma \{ R(i,j') \mid (i,j') \in \mathcal{N} \quad j \neq j' \} = Y \leq Y_0$$

$$\Sigma \{ R(i',j) \mid (i',j) \in \mathcal{N} \quad i' \neq i \} = Z \leq Z_0$$

$$\Sigma \{ R(i',j') \mid (i',j) \in \mathcal{N} \quad i' \neq i, j' \neq j \} = t \leq T_0$$

alors la condition de convergence pour ce point s'écrira :

$$K_R \left[(1+N_y) \frac{b_R + X_0 + Z_0}{c + X_0 + Z_0} + (1+N_x) \frac{a_R + X_0 + Y_0}{c + X_0 + Y_0} + (2N_t - N) \frac{(a_R + X_0 + Y_0)(b_R + X_0 + Z_0)}{(c + X_0 + Y_0)(c + X_0 + Z_0)} \right] < 1 \quad *$$

c'est ce système de conditions que nous avons utilisé dans le cas où les colonnes et les lignes sont toutes distinctes.

2- Algorithme n° 2

Cet algorithme donne une reconstitution d'ordre 0, c'est à dire

$$\forall (i,j) \in \mathcal{N} \qquad K_{ij}^r = 1$$

On a donc

$$\forall (i,j) \in \mathcal{N} \qquad R_{ij}^p = \frac{R_i^{p-1} R_j^{p-1}}{R^{p-1}}$$

a) 1 trou

Ce cas a été étudié précédemment .

C'est d'ailleurs cette étude qui a permis la détermination de cet algorithme

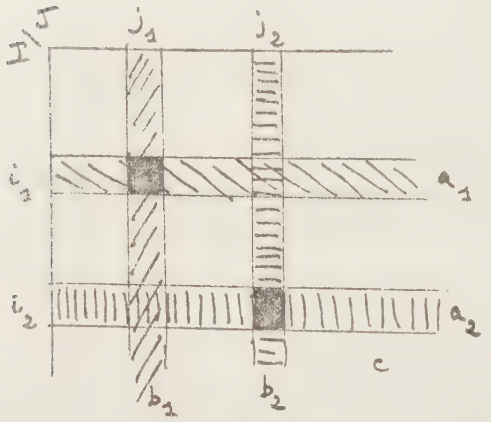
b) 2 trous

A titre d'exercice, considérons le cas particulier où $\mathcal{N}$ se réduit à 2 éléments (i_1, j_1) et (i_2, j_2)

---

* Si $(2N_t - N) < 0$ remplacer la majoration du dernier terme par une minoration, par exemple : $(2N_t - N) \frac{a_h b_h}{(c + T_0)^2}$

• Supposons d'abord $i_1 \neq i_2$ et $j_1 \neq j_2$



Pour simplifier l'écriture, notons

$$a_1 = k^\circ_{i_1} = \Sigma\{ k_{i_1 j} \mid j \in J - \{j_2\} \}$$

$$a_2 = k^\circ_{i_2} = \Sigma\{ k_{i_2 j} \mid j \in J - \{j_2\} \}$$

$$b_1 = k^\circ_{j_1} = \Sigma\{ k_{i j_1} \mid i \in I - \{i_2\} \}$$

$$b_2 = k^\circ_{j_2} = \Sigma\{ k_{i j_2} \mid i \in J - \{i_2\} \}$$

$$c = k^\circ = \Sigma\{ k_{ij} \mid (i,j) \in I \times J - \mathcal{P} \}$$

les suites u_n et v_n sont définies de la façon suivante :

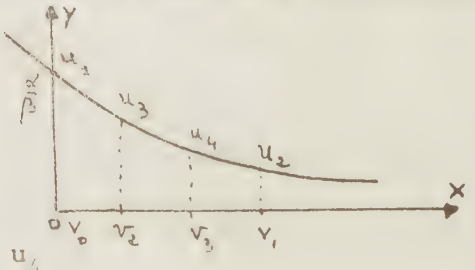
$$u_0 = 0 \qquad u_n = \frac{a_1 b_1}{c - a_1 - b_1 + v_{n-1}}$$

$$v_0 = 0 \qquad v_n = \frac{a_2 b_2}{c - a_2 - b_2 - u_{n-1}}$$

Cela entraîne $u_2 > u_1$

$v_2 > v_1$

La fonction $y = \frac{\alpha}{\beta + x}$ est croissante lorsque x décroit, décroissante lorsque x croît. Ceci entraîne



$u_1 < u_3 < u_2$

$v_1 < v_3 < v_2$

$u_1 < u_3 < u_4 < u_2$ car v_3 détermine u_4

et v_1 détermine u_2.

De même $v_1 < v_3 < v_4 < v_2$, on obtient ainsi deux suites u_{2p}, u_{2p+1} l'une croissante, l'autre décroissante, l'une majorant l'autre. Elles sont donc convergentes. De même les suites v_{2p} et v_{2p+1} sont convergentes.

Si la suite u_{2p} tend vers ℓ, la suite v_{2p+1} vers ℓ' nous devons avoir

$$\ell = \frac{a_1 b_1}{c - (a_1 + b_1) + \ell'} \qquad \ell' = \frac{a_2 b_2}{c - (a_2 + b_2) + \ell}$$

D'où

$$\ell' = \frac{a_1 b_1 - \ell\,(c - (a_1 + b_1))}{\ell} = \frac{a_2 b_2}{c - (a_2 + b_2) + \ell}$$

$$\ell^2(c-(a_1+b_1)) + \ell\left[(c-(a_1+b_1))(c-(a_2+b_2)) - a_1b_1 + a_2b_2\right] - a_1b_1(c-(a_2+b_2)) = 0$$

Or cette équation n'admet qu'une seule solution positive le produit des racines étant négatif puisque l'on a

$$c - (a_2 + b_2) > 0$$

$$c - (a_1 + b_1) > 0$$

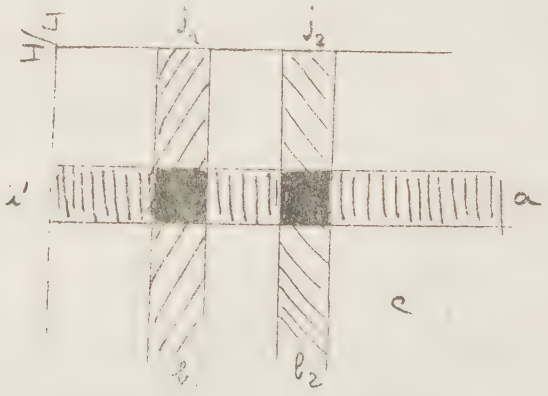
De même, si on considére la suite u_{2p+1} tendant vers L, et la suite v_{2p} tendant vers L', on obtient la même équation en L admettant une seule solution positive.

Autrement dit on doit avoir $\ell = L \quad \ell' = L'$.

Les suites u_n, v_n sont donc convergentes et l'algorithme 2 est bien convergent dans ce cas.

- Supposons maintenant que les "trous" soient dans une même colonne ou une même ligne par exemple $i_1 = i_2 = i'$.

$$j_1 \neq j_2$$



De même, posons

$$a = k_i^\circ = \sum\{k_{ij} \mid j \in J - \{j_1, j_2\}\}$$

$$b_1 = k_{j_1}^\circ = \sum\{k_{ij_1} \mid i \in I - \{i'\}\}$$

$$b_2 = k_{j_2}^\circ = \sum\{k_{ij_2} \mid i \in I - \{i'\}\}$$

$$c = k^\circ = \sum\{k_{ij} \mid (i,j) \in I \times J - \mathcal{C}\}$$

On a alors les suites définies de la façon suivante :

$$u_0 = 0 \qquad v_0 = 0$$

$$u_n = \frac{(a + v_{n-1})\,b_2}{c - (a + b_1)} \qquad v_n = \frac{(a + u_{n-1})\,b_2}{c - (a + b_2)}$$

D'où $$u_n = \frac{1}{c-(a+b_1)}\left[ab_1 + b_1\frac{(a+u_{n-2})b_2}{c-(a+b_2)}\right]$$

$$= \frac{1}{[c-(a+b_1)][c-(a+b_2)]}\left[ab_1(c-a) + b_1b_2u_{n-2}\right]$$

De même

$$v_n = \frac{1}{[c-(a+b_1)][c-(a+b_2)]}\left[ab_2(c-a) + b_1b_2v_{n-2}\right]$$

Considérons la fonction $y = \alpha x + \beta = h(x)$

Posons $x_0 = \gamma$ et $x_n = h(x_{n-1})$

Alors $x_n = \beta\left(\alpha^{n-1} + \alpha^{n-2} + \dots + 1\right) + \alpha^n\gamma$

On aura une suite convergente si et seulement si $|\alpha| < 1$
et sa limite ne dépendra pas de γ

dans ce cas $$\alpha = \frac{b_1b_2}{(c-(a+b_1))(c-(a+b_2))}$$

on sait que $$c-(a+b_1+b_2) > 0$$

soit $$\frac{b_1}{c-a-b_1} < 1$$

et $$\frac{b_2}{c-a-b_2} < 1$$

La suite u_n (resp v_n) est convergente, puisque les suites u_{2p} et u_{2p+1} (resp. v_{2p} et v_{2p+1}) sont convergentes et ont même limite indépendante de la valeur de départ u_o ou u_1 (resp v_o ou v_1)

On aurait pu l'écrire de façon matricielle :

$$\begin{pmatrix} u_n \\ v_n \end{pmatrix} = \begin{pmatrix} 0 & \frac{b_1}{c-a-b_1} \\ \frac{b_2}{c-a-b_2} & 0 \end{pmatrix}\begin{pmatrix} u_{n-1} \\ v_{n-1} \end{pmatrix} + \begin{pmatrix} \frac{ab_1}{c-a-b_1} \\ \frac{ab_2}{c-a-b_2} \end{pmatrix}$$

Soit $X_n = T_0 X_{n-1} + C$

avec $X_n = \begin{pmatrix} u_n \\ v_n \end{pmatrix}$

$$T = \begin{pmatrix} 0 & \frac{b_1}{c-a-b_1} \\ \frac{b_2}{c-a-b_2} & 0 \end{pmatrix}$$

$$C = \begin{pmatrix} \frac{ab_1}{c-a-b_1} \\ \frac{ab_2}{c-a-b_2} \end{pmatrix}$$

$$X_0 = \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

On a alors

$$X_n = T_0 X_{n-1} + C$$
$$X_{n-1} = T_0 X_{n-2} + C$$
$$\vdots$$
$$X_1 = C$$

D'où $X_n = C\,(1 + \cdots T^{n-1})$

Or la matrice T est diagonalisable avec 2 valeurs propres égales et telles que

$$\lambda^2 = \frac{b_1 b_2}{(c-a-b_2)(c-a-b_1)}$$

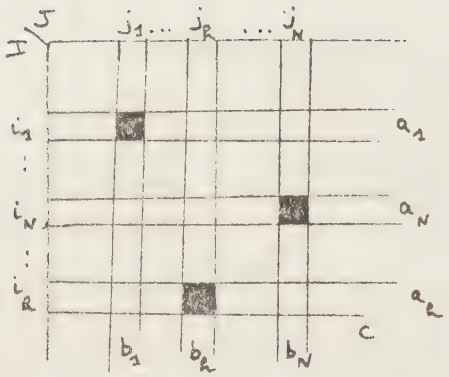
Autrement dit $|\lambda| < 1$. On obtient bien ainsi le même résultat.

$$X_n \to X = \frac{C}{1-T}$$

c- N "trous"

• Supposons que les N "trous" soient dans des colonnes et des lignes distinctes.

$\forall k, l \in [1, N]$ $\quad k \neq l \iff (i_k \neq i_l)$ et $(j_k \neq j_l)$



Comme précédemment, introduisons les notations

$$\forall \ell \in [1,N] \qquad a_\ell = \sum \{ k_{i_\ell j} \mid j \in J - \{j_\ell\} \}$$

$$b_\ell = \sum \{ k_{i j_\ell} \mid i \in I - \{i_\ell\} \}$$

$$c = \sum \{ k_{ij} \mid (i,j) \in I \times J - \mathcal{K} \}$$

On obtient alors N suites $u_{\ell,n}$ définies de la façon suivante :

$$u_{\ell,0} = 0$$

$$u_{\ell,n} = \frac{a_\ell b_\ell}{c - (a_\ell + b_\ell) + \sum_{h \neq \ell} u_{h,n-1}}$$

On peut remarquer, comme dans le cas particulier (N=2) précédent, que si

$$\forall \ell \in [1,N] \qquad u_{\ell,n} \geq u_{\ell,n-1} \quad \Rightarrow \quad \forall \ell \in [1,N] \qquad u_{\ell,n+1} \leq u_{\ell,n}$$

De là, on en déduit 2N suites telles que

$\forall \ell \in [1,N]$ $\quad u_{\ell,2p}$ est une suite croissante

$\quad u_{\ell,2p+1}$ est une suite décroissante

et ces 2N suites telles que l'une majore l'autre sont convergentes.

Il reste à démontrer que ces suites $u_{1,2p}$ et $u_{1,2p+1}$ convergent deux par deux vers la même limite; mais de toute façon, même si les limites des 2 suites étaient distinctes, on obtiendrait par l'algorithme à la limite pour chaque trou un nombre ℓ puis un nombre ℓ' puis ℓ de nouveau ...., autrement dit oscillation.

Or $$u_{h,n} = \frac{a_h b_h}{c - (a_h + b_h) + \sum_{\substack{\ell=1 \\ \ell \neq h}}^{N} u_{\ell,n-1}}$$

$$u_{h,2p+1} - u_{h,2p} = \frac{a_h b_h}{\left[c - (a_h + b_h) + \sum_{\ell \neq h} u_{\ell,2p}\right]\left[c - (a_h + b_h) + \sum_{\ell \neq h} u_{\ell,2p-1}\right]} \left[\sum_{\ell \neq h} u_{\ell,2p-1} - \sum_{\ell \neq h} u_{\ell,2p}\right]$$

$$u_{h,2p-1} - u_{h,2p} = \frac{a_h b_h}{\left[c - (a_h + b_h) + \sum_{\ell \neq h} u_{\ell,2p}\right]\left[c - (a_h + b_h) + \sum_{\ell \neq h} u_{\ell,2p-1}\right]} \left[\sum_{\ell \neq h} u_{\ell,2p-1} - \sum_{\ell \neq h} u_{\ell,2p-2}\right]$$

$$u_{h,2p+1} - u_{h,2p} < \frac{a_h b_h}{\left[c - (a_h + b_h)\right]^2} \left[\sum_{\ell \neq h} (u_{\ell,2p-1} - u_{\ell,2p})\right]$$

Posant $$\alpha = \sum_{k=1}^{N} \frac{a_k b_k}{c-(a_k+b_k)}$$

$$\beta = \max \left\{ \sum_{k=1}^{N} \frac{a_k b_k}{(c-(a_k+b_k))^2} - \frac{a_h b_h}{(c-(a_h+b_h))^2} \mid h \in [1,N] \right\}$$

$$= \sum_{k=1}^{N} \frac{a_k b_k}{[c-(a_k+b_k)]^2} - \min \left\{ \frac{a_h b_h}{[c-(a_h+b_h)]^2} \mid h \in [1,N] \right\}$$

$\Rightarrow$

$$u_{h,1} - u_{h,0} < \alpha$$

$$u_{h,1} - u_{h,2} < \alpha\beta$$

$$\vdots$$

$$u_{h,2p+1} - u_{h,2p} < \alpha \beta^{2p}$$

si $\beta < 1$ $\qquad |u_{h,2p+1} - u_{h,2p}| \to 0$

et les suites $u_{h,2p}$, $u_{h,2p+1}$ convergent vers la même limite.

On obtient donc la condition de convergence :

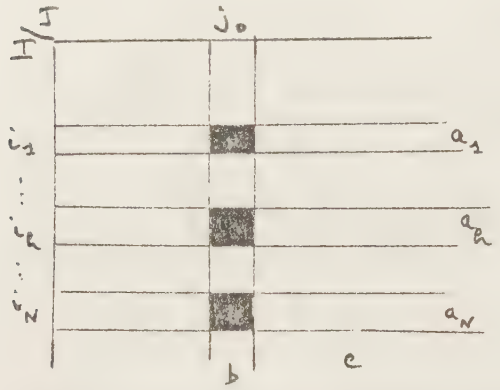
$$\sum_{k=1}^{N} \frac{a_k b_k}{[c-(a_k+b_k)]^2} < 1 + \min \left\{ \frac{a_h b_h}{[c-(a_h+b_h)]^2} \mid h \in [1,N] \right\}$$

On peut remarquer que la condition de convergence de l'algorithme 1 dans le cas d'une approximation d'ordre 0 donne alors :

$$\forall h \in [1,N] \qquad [c-(a_h+b_h)]^2 > (N-1)\, a_h b_h$$

Ce système de conditions est plus restrictif que la condition précédente puisque celle-ci en découle.

- Supposons que tous les "trous" soient dans une même colonne ou une même ligne (par exemple j_o)



De même notons

$$c = \sum \{ k_{ij} \mid (i,j) \in I \times J - \mathcal{H} \}$$

$$b = \sum \{ k_{i j_o} \mid i \in I - \{ i_1, \dots i_N \} \}$$

$$a_k = \sum \{ k_{i_k j} \mid j \in J - \{ j_o \} \}$$

On obtient les suites définies par :

$$u_{\ell,n} = 0$$

$$u_{\ell,n} = \frac{a_\ell \left(b + \sum_{k \neq \ell} u_{k,n-1} \right)}{c - (a_\ell + b)}$$

On voit que ces suites sont croissantes puisque pas à pas tous les $u_{p,n-1}$ croissent.

soit

$$a_h = \inf \{ a_\ell \mid \ell \in [1,N] \}$$

$$\frac{a_h}{c-b-a_h} = \inf \left\{ \frac{a_\ell}{c-b-a_\ell} \mid \ell \in [1,N] \right\}$$

$$u_{\ell,n} = \frac{a_\ell}{c-(a_\ell+b)} \left(b + \sum_{p \neq \ell} u_{p,n-1} \right)$$

Posons $$\gamma = \sum_{\ell=1}^{N} \frac{a_\ell}{c-(a_\ell+b)} - \frac{a_h}{c-(a_h+b)}$$

$$u_{\ell,1} = \frac{a_\ell}{c-(b+a_\ell)}$$

$$u_{\ell,2} < \frac{a_\ell}{c-(b+a_\ell)} \left(b + \gamma \right)$$

$$\vdots$$

$$u_{\ell,n} < \frac{a_\ell}{c-(b+a_\ell)} \left(\gamma^{n-1} + (1 + \dots + \gamma^{n-2}) b \right)$$

La suite γ_n définie par

$$\gamma_0 = 0$$
$$\gamma_1 = \gamma$$
$$\gamma_n = \frac{a_\ell}{c-(b+a_\ell)} \left(\gamma^{n-1} + (1+\dots+\gamma^{n-2})b \right)$$

est convergente si $|\gamma| < 1$

soit si $$\frac{a_1}{c-(b+a_1)} + \dots + \frac{a_N}{c-(b+a_N)} - \frac{a_h}{c-(b+a_h)} < 1$$

$$\frac{a_1}{c-(b+a_1)} + \dots + \frac{a_N}{c-(b+a_N)} < 1 + \min \left\{ \frac{a_\ell}{c-(b+a_h)} \mid h \in [1,N] \right\}$$

les suites $u_{1,n}$ seront convergentes si

$$\sum \left\{ \frac{a_\ell}{c-(b+a_\ell)} \mid h \in [1,N] \right\} < 1 + \min \left\{ \frac{a_\ell}{c-(b+a_\ell)} \mid h \in [1,N] \right\}$$

Il est possible d'étudier les suites $u_{1,n}$ de façon matricielle.

Posons $$U_n = \begin{bmatrix} u_{1,n} \\ \\ u_{N,n} \end{bmatrix} \qquad T = \begin{bmatrix} 0 & \gamma_1 & \dots & \gamma_1 \\ \gamma_2 & 0 & \dots & \gamma_2 \\ & & \ddots & \\ \gamma_N & & & 0 \end{bmatrix} \qquad B = \begin{bmatrix} \gamma_1 b \\ \vdots \\ \gamma_N b \end{bmatrix}$$

Cela donne

$$U_n = T \circ U_{n-1} + B$$
$$U_n = T^n \circ U_0 + B(1+\dots+T^{n-1})$$

Soient E et F deux matrices quelconques N x N, $E = [e_{h\ell}]$

Sur l'ensemble des matrices N x N considérons la norme définie par

$$\|E\| = \max \left\{ \sum_{\ell=1}^{N} |e_{h\ell}| \mid h \in [1,N] \right\}$$

Ceci entraîne

$$\forall h,\ell \in [1,N] \qquad |e_{h\ell}| \leq \|E\|.$$

Cette norme possède les propriétés

$$\|E+F\| \leq \|E\| + \|F\|$$
$$\|\lambda E\| = |\lambda| \, \|E\|$$
$$\|E.F\| \leq \|E\| \cdot \|F\|$$
$$\|E^n\| \leq (\|E\|)^n$$

Si γ_{hl}^{n} est un élément de T^n,

$$|\gamma_{hl}^{n}| \leq \|T^n\|$$

On obtiendra une condition suffisante de convergence si $\|T\|<1$, soit si

$$\max\left\{ \sum\left\{ \frac{a_h}{c-(a_h+b)} \mid h\in[1,N], h\neq l\right\} \mid l\in[1,N]\right\} < 1$$

soit si

$$\sum_{h=1}^{N} \frac{a_h}{c-(a_h+b)} < 1 + \min_{h\in[1,N]} \frac{a_h}{c-(a_h+b)}$$

ceci redonne bien la condition précédente.

En opérant sur les lignes et non sur les colonnes, on obtiendrait une condition plus restrictive

$$\forall h\in[1,N] \qquad (N-1)\frac{a_h}{c-(a_h+b)} < 1$$

• $\mathcal{N}$ quelconque

On appliquera la méthode utilisée dans le cas général pour l'algorithme 1. Considérons la fonction G(y,z,t)

$$G(y,z,t) = \frac{(a+y)(b+z)}{c-(a+b)+t}$$

En utilisant les mêmes notations, on obtient

$$\Delta = \left|\frac{\partial G}{\partial x}\right| + N_y\left|\frac{\partial G}{\partial y}\right| + N_z\left|\frac{\partial G}{\partial z}\right| + N_t\left|\frac{\partial G}{\partial t}\right| = N_y\frac{\partial G}{\partial y} + N_z\frac{\partial G}{\partial z} - N_t\frac{\partial G}{\partial t}$$

puisque y,z,t sont des sommes respectives de N_y, N_z, N_t inconnues

D'où

$$\Delta = N_y\frac{b+z}{c-(a+b)+t} + N_z\frac{a+y}{c-(a+b)+t} + N_t\frac{(a+y)(b+z)}{[c-(a+b)+t]^2}$$

Dans le cas où x,y,z,t, appartiennent à des intervalles finis, on obtiendra un système de conditions suffisant de convergence en écrivant pour chaque point $\Delta<1$

En particulier, si l'algorithme ITER donne deux valeurs successives (x_1,y_1,z_1t_1) et (x_2,y_2,z_2,t_2), il suffira de vérifier l'existence d'un nombre γ ($\gamma>1$) pour lequel le système de conditions

$\Delta < \frac{1}{\gamma}$ est vérifié avec $X = x_1 + \frac{\gamma}{\gamma-1}|x_1-x_2|$

3- Algorithme n° 3

Nous avons vu au chapitre III-3 que cet algorithme se construisait par récurrence sur le nombre n des N trous, en utilisant pour le premier trou l'algorithme 2.

Supposons que tous les trous se trouvent dans des lignes et des colonnes distinctes : $\forall k,l \in [1,N] \quad k \neq l \iff \begin{cases} i_l \neq i_k \\ j_l \neq j_k \end{cases}$

Notons a_k, b_k, c les sommes partielles comme précédemment.

Posons l'hypothèse de récurrence suivante :

Pour (n-1) points, l'algorithme converge et la somme de ces (n-1) valeurs est une fonction décroissante de c.

Notons cette somme v(c).

Pour passer à n trous, posons :

$$u_0 = \frac{a_n b_n}{c-(a_n+b_n)}$$ la valeur calculée pour le n^{e} trou.

Cette valeur change le tableau précédent :

$$c \rightarrow c+u_0$$

On obtient alors

$$v_1 = v(c+u_0) \quad \Rightarrow \quad v_1 < v(c)$$

Posant

$$u_1 = \frac{a_n b_n}{c-(a_n+b_n)+v_1} \quad \Rightarrow \quad 0 < u_1 < u_0$$

On considère

$$v_2 = v(c+u_1) \qquad v_1 < v_2 < v(c)$$

$$u_2 = \frac{a_n b_n}{c-(a_n+b_n)+v_2} \qquad 0 < u_2 < u_1$$

.....

On voit qu'on arrive ainsi à

$$v_p = v(c+u_{p-1})$$

$$u_p = \frac{a_n b_n}{c-(a_n+b_n)+v_p}$$

On obtient la suite u_n décroissante minorée donc convergente. Et pour cette limite u, on obtient une solution $v(c+u)$. La convergence existe bien pour n points.

De plus, soit

$$\Delta = u + v = \frac{a_n b_n}{c-(a_n+b_n)+v} + v$$

$$\frac{d\Delta}{dc} = \frac{dv}{dc}\left[1 - \frac{a_n b_n}{[c-(a_n+b_n)+v]^2}\right]$$

or $\frac{dv}{dc}$ est négatif par hypothèse.

Notamment si $[c-(a_n+b_n)+v]^2 - a_n b_n > 0$ on a bien une solution décroissante lorsque c croit.

L'hypothèse de récurrence étant juste pour n = 1,

on obtient la convergence en particulier si

$$\forall h \in [1,N] \qquad [c-(a_h+b_h)]^2 > a_h b_h$$

• Supposons maintenant que tous les trous se trouvent dans une même colonne (ou ligne)

Notons de la même façon a, b_h, c.

Faisons l'hypothèse de récurrence suivante en supposant les b_h rangés en ordre décroissant : $b_1 \geqslant b_2 \cdots \geqslant b_N$:

Pour (n-1) points, la convergence existe et la somme de ces (n-1) valeurs est une fonction v telle que v croisse lorsque a et c croisse d'un même nombre x. On suppose de plus que

$$\frac{dv}{dx} < \frac{c-(a+b_{n-1})}{b_{n-1}}$$

Pour le $n^{ème}$ trou, nous avons

$$v_0 = v(0)$$

$$u_0 = \frac{b_n\ (a + v(0))}{c - (a + b_n)}$$

$$v_1 = v(u_0) \quad \Rightarrow \quad v_0 < v_1$$

$$u_1 = \frac{b_n\ (a + v_1)}{c - (a + b_n)}$$

. . . . .

$$v_p = v(u_{p-1})$$

$$u_p = \frac{b_n\ (a + v_p)}{c - (a + b_n)}$$

On obtient des suites croissantes v_p et u_p

$$u_p - u_{p-1} = \frac{b_n}{c - (a + b_n)} (v_p - v_{p-1})$$

comme $\frac{dV}{dx} < \alpha$, on a

$$u_p - u_{p-1} = \frac{b_n}{c - (a + b_n)} (u_{p-1} - u_{p-2})\, v'(\theta) \leq \frac{\alpha\, b_n}{c - (a + b_n)} (u_{p-1} - u_{p-2})$$

$$u_p - u_{p-1} \leq \left[\frac{b_n\, \alpha}{c - (a + b_n)}\right]^{p-1} (u_1 - u_0)$$

On aura donc convergence des suites u_p et v_p lorsque $\frac{b_n\, \alpha}{c - (a + b_n)} < 1$

Or $\alpha < \frac{c - (a + b_{n-1})}{b_{n-1}}$

$$\Rightarrow \quad \frac{b_n\, \alpha}{c - (a + b_n)} < \frac{b_n}{c - (a + b_n)} \times \frac{c - (a + b_{n-1})}{b_{n-1}} \leq 1$$

Car $\frac{b_n}{c - (a + b_n)} \leq \frac{b_{n-1}}{c - (a + b_{n-1})}$

De plus, $u = \dfrac{b_n(a+v)}{c-(a+b_n)}$

$v = v(u)$

$\Rightarrow \quad u(c-(a+b_n)) = b_n(a+v)$

$u+v = \dfrac{c-a}{b_n}u - a$

$\dfrac{d(u+v)}{dx} = \dfrac{c-a}{b_n}\dfrac{du}{dx} - 1 = v'\left(1+\dfrac{du}{dx}\right) + \dfrac{du}{dx}$

$\dfrac{du}{dx} = \dfrac{(1+v')\,b_n}{c-(a+b_n)-b_n v'}$

$\dfrac{v' b_n}{c-(a+b_n)} < 1 \quad \Rightarrow \quad \dfrac{du}{dx} > 0$

$\dfrac{d(u+v)}{dx} = \dfrac{v'(c-(a+b_n))+b_n}{c-a-b_n(1+v')}$

$\dfrac{d(u+v)}{dx} < \dfrac{(1+2h_n)/h_{n-1} + h_n}{1-h_n/h_{n-1}}$ en posant $h_n = \dfrac{b_n}{c-(a+b_n)}$

$h_{n-1} = \dfrac{b_{n-1}}{c-(a+b_{n-1})}$

$\dfrac{d(u+v)}{dx} < \dfrac{1+2h_n+h_n h_{n-1}}{h_{n-1}-h_n}$ avec $h_n \le h_{n-1} < 1$

$\dfrac{d(u+v)}{dx} - \dfrac{1}{h_n} < \dfrac{h_n^2(2+h_{n-1}) + 2h_n - h_{n-1}}{h_{n-1}-h_n}$

Si nous étudions le binôme en x

$B(x) = x^2(2+h) + 2x - h$

$B(0) < 0$

$B(1) = 0$

Donc $B(x) < 0$ pour $0 < x < 1$

$\Rightarrow \quad \dfrac{d(u+v)}{dx} < \dfrac{1}{h_n} = \dfrac{c-(a+b_n)}{b_n}$

Ceci démontre bien l'hypothèse de récurrence puisque pour le premier point :

$$v_1 = \frac{a\,b_1}{c-(a+b_1)} \quad \Rightarrow \quad \frac{dv_1}{dx} = \frac{b_1}{c-(a+b_1)}$$

le second point s'obtiendra de la même façon :

$$\frac{b_1}{c-(a+b_1)} \times \frac{b_2}{c-(a+b_2)} < 1 \quad \text{puisque} \quad \begin{matrix} c-(a+b_1) > b_2 \\ c-(a+b_2) > b_1 \end{matrix}$$

Pour lui, on aura alors

$$\frac{dV_2}{dx} < \frac{c-(a+b_2)}{b_2}$$

Si nous prenons les trous dans l'ordre des (b_h) décroissants, l'algorithme sera convergent.

4- L'algorithme 4

- Supposons que tous les "trous" se trouvent dans des lignes et des colonnes distinctes Notons de la même façon a_h, b_h, c les sommes partielles initiales, et K_h le terme correctif correspondant au h^{e} trou.

Faisons l'hypothèse de récurrence suivante :

Pour (n-1) points, la convergence existe et la somme de ces (n-1) valeurs est une fonction décroissante de c soit $v(c)$ avec $-1 < v'(c) < 0$ (i.e c+v est une fonction croissante de c).

Posons de même $v_0 = v(c)$

$$u_1 = K\,\frac{a_n b_n}{c+v_0}$$

$$v_1 = v(c+u_1) \qquad v_1 < v_0$$

$$u_2 = K_n\,\frac{(a_n+u_1)(b_n+u_1)}{c+u_1+v_1}$$

$$v_2 = v(c+u_2)$$

d'après ce qu'on a vu précédemment, la fonction $y = \frac{(a+x)(b+x)}{c+x}$ est une fonction croissante de x, pour x positif.

Cela entraîne : $u_2 > u_1$

$v_2 < v_1$

.....

De cette façon, on définit une suite u_n croissante et une suite v_n décroissante.

La suite v_n ,minorée par 0, tend vers une limite v , v(c) étant une fonction strictement décroissante de c, la suite u_n est convergente sinon la suite v_n ne pourrait être convergente.

On peut remarquer que cette convergence exige la condition suivante

$$K_n < \frac{c+v}{a_n+b_n}$$

et

$$\left[c+v-K_n(a_n+b_n)\right]^2 - 4K_n(K_n-1)a_n b_n \geq 0$$

Considérons la suite u'_n définie par :

$$u'_1 = K_n \frac{a_n b_n}{c}$$

et

$$u'_n = K_n \frac{(a_n+u'_{n-1})(b_n+u'_{n-1})}{c+u'_{n-1}}$$

Cette suite est une suite majorant la suite u_n : $u_1 < u'_1$

et si $u_{n-1} < u'_{n-1}$, cela exige

$$u_n < K_n \frac{(a_n+u_{n-1})(b_n+u_{n-1})}{c+u_{n-1}} < K_n \frac{(a_n+u'_{n-1})(b_n+u'_{n-1})}{c+u'_{n-1}}$$

$$u_n < u'_n$$

Cette suite u'_n est convergente si on a les conditions :

$$\begin{cases} K_n < \dfrac{c}{a_n+b_n} \\ \left[c-K_n(a_n+b_n)\right]^2 - 4K_n(K_n-1)a_n b_n \geq 0 \end{cases}$$

Or ces 2 conditions entraînent bien les 2 conditions précédentes. Pour établir la récurrence, il reste à démontrer que

$$-1 < \frac{d(u+v)}{dc} < 0$$

On a les formules suivantes :

$$\begin{cases} u = K_n \dfrac{(a_n+u)(b_n+u)}{c+u+v} \\ v = v(c+u) \end{cases}$$

$$\Rightarrow \quad \frac{dv}{dc} = v'\left(1+\frac{du}{dc}\right)$$

$$\frac{d(u+v)}{dc} = (v'+1)\frac{du}{dc} + v'$$

$$\frac{d(u+v+c)}{dc} = (1+v')\left(\frac{du}{dc}+1\right)$$

De ces deux dernières relations, puisque $-1 < v' < 0$, on déduit immédiatement que

$$-1 < \frac{du}{dc} < 0 \qquad \Rightarrow \quad -1 < \frac{d(u+v)}{dc} < 0$$

Or

$$u(c+u+v) = K_m (a_m+u)(b_m+u)$$

$$\left[2u(1-K_m) + (c+v-K_m(a_m+b_m))\right]\frac{du}{dc} + u\left(1+\frac{dv}{dc}\right) = 0$$

$$\left[2u(1-K_m) + (c+v-K_m(a_m+b_m))\right]\frac{du}{dc} + u\left(1+v'+v'\frac{du}{dc}\right) = 0$$

$$\frac{du}{dc} = \frac{-u(1+v')}{2u(1-K_m)+uv'+c+v-K_m(a_m+b_m)}$$

$$-1 < \frac{du}{dc} < 0 \qquad \Leftrightarrow \qquad -1 < \frac{-u(1+v')}{2u(1-K_m)+uv'+(c+v-K_m(a_m+b_m))} < 0$$

En tenant compte du fait que $-1 < v' < 0$

$$2u(1-K_m u) + uv' + c+v-K_m(a_m+b_m) > 0$$

$$u(1-2K_m) + u(1+v') + c+v-K_m(a_m+b_m) > 0$$

et

$$u(1+v') < u(1-2K_m) + u(1+v') + c+v-K_m(a_m+b_m)$$

Donc si

$$u(2K_m-1) < c+v-K_m(a_m+b_m)$$

la fonction (u + v) est une fonction décroissante de c et la fonction $(u+v+c)$ est une fonction croissante de c.

si $1-2K_m \geq 0$, cette condition est toujours satisfaite puisqu'alors

$$c > a_m+b_m \geq K_m(a_m+b_m)$$

Cette équation peut alors s'écrire, en supposant $K_m \geq \frac{1}{2}$

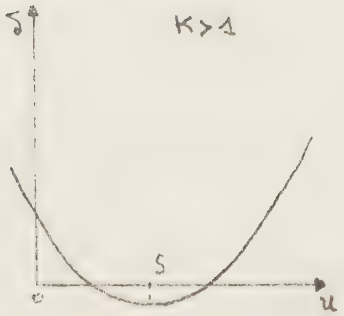
$$u < \frac{c+v-K_m(a_m+b_m)}{2K_m-1}$$

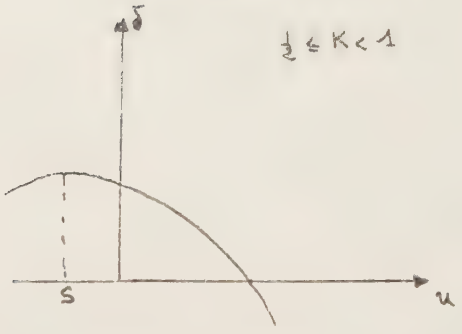
Cela exige notamment :

$$c+v-K_m(a_m+b_m) > 0$$

$$K_m < \frac{c+v}{a_m+b_m}$$

ce qui est la première des deux conditions que nous avons trouvées nécessaires précédemment.





Or u est tel que

$$u = K_n \frac{(a_n+u)(b_n+u)}{c+u+v}$$

soit

$$\delta(u) = u^2(K_n-1) + u(K_n(a_n+b_n) - c + v) + K_n a_n b_n = 0$$

les racines u_1 et u_2 sont telles que

$$S = \frac{u_1+u_2}{2} = \frac{c+v-K_n(a_n+b_n)}{2(K_n-1)}$$

Si $K>1$ on a $\frac{c+v-K_n(a_n+b_n)}{2K_n-1} < S$

et $u < S$ car $u < \frac{c-K_n(a_n+b_n)}{2(K_n-1)}$

d'après la majoration de u_n :

$$u_n < K_n \frac{(a_n+u_n)(b_n+u_n)}{c+u_n}$$

Si $K<1$ on a $S<0$

Dans ces deux cas, on doit avoir $\delta\left(\frac{c+v-K_n(a_n+b_n)}{2K_n-1}\right) \le 0$

pour obtenir $u < \frac{c+v-K_n(a_n+b_n)}{2K_n-1}$

Soit

$$(K_n-1)\left(\frac{c+v-K_n(a_n+b_n)}{2K_n-1}\right)^2 + \frac{c+v-K_n(a_n+b_n)}{2K_n-1}\left(K_n(a_n+b_n)-c-v\right) + K_n a_n b_n \le 0$$

$$\frac{K_n}{(2K_n-1)^2}\left[(c+v-K_n(a_n+b_n))^2 - a_n b_n (2K_n-1)^2\right] \ge 0$$

$$\left[c+v-K_n(a_n+b_n)\right]^2 - a_n b_n (2K_n-1)^2 \ge 0$$

pour $K = 1$, $u = \frac{a_n b_n}{c+v-(a_n+b_n)}$

Donc $\frac{a_n b_n}{c+v-(a_n+b_n)} < c+v-(a_n+b_n)$

on retrouve l'inéquation précédente avec $K_n = 1$:

$$\left[c+v-(a_n+b_n)\right]^2 > a_n b_n$$

Il faut remarquer que cette condition est la même que celle trouvée pour l'algorithme 3 étudié précédemment.

De même la condition

$$\left[c+v-K_n(a_n+b_n)\right]^2 - a_n b_n (2K_n-1)^2 \geq 0$$

entraîne la vérification de la deuxième condition nécessaire précédente.

Dans le cas :

$$K_n < \frac{v+c}{a_n+b_n}$$

$$\left[c+v-K_n(a_n+b_n)\right]^2 \geq a_n b_n (2K_n-1)^2$$

on obtient une fonction (u + v) vérifiant la condition de récurrence.
Ces conditions sont notamment vérifiées si :

$$K_n < \frac{c}{a_n+b_n}$$

$$\left[c-K_n(a_n+b_n)\right]^2 \geq a_n b_n (2K_n-1)^2$$

De plus, pour N = 1, on a bien une solution unique et cette solution x est telle que

$$x = K\frac{(a+x)(b+x)}{c+x}$$

en notant $K = K_1$, $a = a_1$, $b = b_1$

Il reste à démontrer qu'on a bien

$$\frac{dx}{dc} < 0 \quad \text{et} \quad \frac{dx}{dc} + 1 > 0$$

Or ceci ramène au problème que l'on vient de traiter pour le $n^{ème}$ point mais avec v = o.

Autrement dit, cela exige :

$$K < \frac{c}{a+b}$$

$$\left[c-K(a+b)\right]^2 \geq ab\,(2K-1)^2$$

Etudions la condition trouvée pour K_n.

$$K_n < \frac{c}{a_n+b_n}$$

$$\left[c-K_n(a_n+b_n)\right]^2 \geq a_n b_n (2K_n-1)^2$$

Ceci ramène à étudier le système suivant

$$\begin{cases} P(x) = x^2(a+b)^2 - 2xc(a+b) + c^2 - ab(2x-1)^2 \geq 0 \\ x < \frac{c}{a+b} \end{cases}$$

soit $P(x) = x^2(a-b)^2 - 2x(c(a+b)-2ab) + c^2 - ab$

Pour étudier l'équation $P(x) = 0$, calculons le discriminant Δ

$$\Delta = (c(a+b) - 2ab)^2 - (a-b)^2(c^2 - ab)$$
$$= c^2 4ab - 4ab(a+b)c + 4a^2b^2 + ab(a-b)^2$$
$$= ab[2c - (a+b)]^2$$

Δ est toujours positif et $2c - a - b > 0$

$$P\left(\frac{c}{a+b}\right) = \frac{1}{(a+b)^2}\left[c^2(a-b)^2 - 2c(a+b)(c(a+b) - 2ab) + (c^2 - ab)(a+b)^2\right]$$

$$= \frac{1}{(a+b)^2}\left[-c^2 4ab + 4ab(a+b)c - ab(a+b)^2\right]$$

$$= -ab\,\frac{(2c-a-b)^2}{(a+b)^2}$$

Supposons $a \neq b$

Soit x_1 la plus petite racine de $P(X) = 0$, x_1 est positive, la somme et le produit des racines étant tous deux positifs. Le système d'équations est alors équivalent à

$$x \leq x_1$$

avec

$$x_1 = \frac{c(a+b) - 2ab - \sqrt{\Delta}}{(a-b)^2}$$

$$= \frac{c(a+b) - 2ab - 2c\sqrt{ab} + (a+b)\sqrt{ab}}{(a-b)^2}$$

$$= \frac{c(\sqrt{a} - \sqrt{b})^2 + \sqrt{ab}(\sqrt{a} - \sqrt{b})^2}{(a-b)^2}$$

$$= \frac{c + \sqrt{ab}}{(\sqrt{a} + \sqrt{b})^2} = \frac{c + \sqrt{ab}}{a + b + 2\sqrt{ab}}$$

Supposons maintenant que $a = b$

$$P(X) = -4 \times a(c-a) + c^2 - a^2$$

$$P(X) \geq 0 \iff X \leq \frac{c+a}{4a}$$

on retrouve de cette façon la condition précédente avec $a = b$.

L'algorithme est donc convergent si $K_n \leq \dfrac{c + \sqrt{a_n b_n}}{a_n + b_n + 2\sqrt{a_n b_n}}$

* Supposons maintenant que tous les "trous" se trouvent dans une même ligne (ou colonne)

Notons de même a, b_h, c.

Faisons l'hypothèse de recurrence suivante :

Pour (n-1) point la convergence existe et la somme des (n-1) valeurs est une fonction $\nu(x)$ telle que ν croisse lorsque a et c croissent d'un même nombre x.

Pour le n^{e} "trou", nous avons :

$$v_0 = \nu(0)$$

$$u_1 = K_n \frac{b_n(a+v_0)}{c+v_0}$$

$$v_1 = \nu(u_1) \qquad v_0 < v_1$$

$$u_2 = K_n \frac{(a+u_1+v_1)(b_n+u_1)}{c+u_1+v_1}$$

$$\vdots$$

$$v_{p-1} = \nu(u_{p-1})$$

$$u_p = K_n \frac{(a+u_{p-1}+v_{p-1})(b_n+u_{p-1})}{c+u_{p-1}+v_{p-1}}$$

Les suites u_p et v_p sont croissantes puisque ν croit et que $f(x) = K\frac{(a+x)(b+x)}{c+x}$ est une fonction croissante de x.

$$u_p < K_n(b_n+u_{p-1})$$

u_p sera convergente en particulier si $K_n < 1$.

De plus

$$K_n \frac{(a+u_{p-1})(b_n+u_{p-1})}{c+u_{p-1}} < u_p$$

cela entraîne que u_p sera divergent si les conditions suivantes ne sont pas vérifiées :

$$\forall h \in [1,N] \quad \begin{cases} K_h < \dfrac{c}{a_h+b_h} \\ [c-K_h(a_h+b_h)]^2 - 4K_h(K_h-1)a_h b_h \geq 0 \end{cases}$$

Il reste à démontrer que u + v est une fonction croissante lorsque a et c croisse d'un même nombre.

Considérons la façon dont u et v sont déterminés.

D'après l'hypothèse de récurrence et le fait que la fonction $y = K\frac{(a+x)(b+x+y)}{c+x+y}$

est une fonction croissante de y, nous voyons que si a et c croissent d'un même nombre x, on obtiendra deux suites u'_n et v'_n telles que

$$\forall n \qquad \begin{matrix} u_n \leq u'_n \\ v_n \leq v'_n \end{matrix}$$

Autrement dit, aux limites si elles existent, on aura

$$\begin{matrix} u \leq u' \\ v \leq v' \end{matrix} \Rightarrow u+v \leq u'+v'$$

L'hypothèse de récurrence est ainsi démontrée.

5- Equivalence des algorithmes

Les algorithmes 1,4 conduisent lorsqu'ils convergent à un ensemble de solutions du même système de N équations du second degré par rapport à N inconnues.

Lorsque tous les K_h sont égaux à 1, les quatre algorithmes étudiés recherchent une solution de ce même système d'équations.

Les conditions de convergence des algorithmes 3 et 4, moins restrictives dans les cas particuliers considérés, nous assurent donc de l'existence d'une solution au système d'équations considéré alors même que les algorithmes 1 et 2 peuvent être divergents. Dans ce cas, on pourra obtenir une généralisation de ce programme en utilisant une autre méthode de résolution d'équations, par exemple celle de Newton-Raphson.

Dans le tableau suivant, on trouvera en résumé les conditions nécessaires ou suffisantes trouvées pour les 4 algorithmes.

1 point	convergence si et seulement si : $\begin{cases} K < \frac{c}{a+b} \\ [c-K(a+b)]^2 - 4K(K-1)ab \geq 0 \end{cases}$ $\Leftrightarrow K\left[\sqrt{a(c-b)}+\sqrt{b(c-a)}\right]^2 \leq c^2$	convergence vers $\frac{ab}{c-(a+b)}$	= algorithme 2	= algorithme 1
N points lignes et colonnes ≠	suite majorante pour le trou h si $K_h < \frac{c}{a_h+b_h}$ $[c-K_h(a_h+b_h)]^2 - 4K_h(K_h-1)a_hb_h \geq 0$ convergence si $\forall h \in [1,N]$ $[c-K_h(a_h+b_h)]^2 \geq a_hb_hK_h\frac{[2K_h+N-3]^2}{K_h+N-2}$	suites $u_{h,2p}$, $u_{h,2p+1}$ uniformes, bornées. convergence oscillation convergence si $\sum_{h=1}^{N}\frac{a_hb_h}{[c-(a_h+b_h)]^2} < 1+\min_{h\in[1,N]}\frac{a_hb_h}{(a_h+b_h)^2}$	convergence si $\forall h \in [1,N]$ $\frac{a_hb_h}{[c-(a_h+b_h)]^2} < 1$	convergence si $\forall h \in [1,N]$ $\begin{cases} K_h < \frac{c}{a_h+b_h} \\ [c-K_h(a_h+b_h)]^2 \geq a_hb_h(2K_h-1) \end{cases}$ $\Leftrightarrow K_h < \frac{c+\sqrt{a_hb_h}}{a_h+b_h+2\sqrt{a_hb_h}}$
N points même ligne ou même colonne	suites croissantes convergentes majorées si $\forall h \in [1,N]$ $\sqrt{a(K_hc-\sum_{h=1}^{N}K_hb_h)}+\sqrt{\sum_{h=1}^{N}K_hb_h(c-a)} \leq c$ divergentes si $\forall h \in [1,N]$ $\sqrt{a(K_hc-\sum_{h=1}^{N}K_hb_h)}+\sqrt{\sum_{h=1}^{N}K_hb_h(c-a)} > c$	suite $u_{k,n}$ croissantes majorées si $\sum_{h=1}^{N}\frac{b_h}{c-(a+b_h)} < 1+\min_{h\in[1,N]}\frac{b_h}{c-(a+b_h)}$	convergence toujours réalisée en prenant les trous dans l'ordre des b_h décroissants	convergence si $K_h < 1$ divergence si on n'a pas $\forall h \in [1,N]$ $K_h < \frac{c}{a+b_h}$
N points	Suite majorante si $\sum_{h=1}^{N}K_h < \frac{c}{\sup a_h+\sup b_h}$ $[(c-2K_h)(c+\rho_0+\sup b_h)]^2 - 4(2K_h)(2K_h-1)\sup a_h \sup b_h \geq 0$ convergence si $\forall h \in [1,N]$ $K_h\left[(1+N_y)\frac{b_h+X_0+Z_0}{c+X_0+Z_0}+(1+N_3)\frac{a_h+X_0+Y_0}{c+X_0+Y_0}\right.$ $\left.+(2N_\varepsilon-N)\frac{(a_h+X_0+Y_0)(b_h+X_0+Z_0)}{(c+X_0+Y_0)(c+X_0+Z_0)}\right] < 1$ si $2N_\varepsilon - N \geq 0$, sinon ... $(2N_\varepsilon-N)\frac{a_hb_h}{(c+\rho_0)^2}$	convergence si $\forall h \in [1,N]$ $N_y\frac{b_h+Z_0}{c-a_h-b_h}+N_3\frac{a_h+Y_0}{c-a_h-b_h}+$ $+N_\varepsilon\frac{(b_h+Z_0)(a_h+Y_0)}{c-a_h-b_h} < 1$		

V - ETUDE POUR UNE APPROXIMATION DE RANG r

Dans le chapitre précédent, nous avons étudié la convergence d'un algorithme (ITER) utilisé à chaque pas. Il reste à vérifier la convergence de l'algorithme complet pour tout r. Pour une approximation de rang 0, la convergence se réduit à celle de l'étude précédente ($\mathcal{K} = 1$ pour tous les points).

1 - Rappel d'algèbre linéaire

a) Normes de matrices carrées

Soit E un espace vectoriel de dimension n sur $\mathbb{R}$ ($\mathbb{R}^n$ par ex.). On suppose E muni des normes usuelles habituelles.

$\forall x \in E$

$$\|x\|_1 = \sum_{i=1}^{n} |x_i|$$

$$\|x\|_2 = \left[\sum_{i=1}^{n} x_i^2\right]^{1/2}$$

$$\|x\|_\infty = \max \left\{ |x_i| \mid 1 \le i \le n \right\}$$

Ces normes sont équivalentes et nous avons

$\forall x \in E$

$$\|x\|_\infty \le \|x\|_1$$
$$\|x\|_\infty \le \|x\|_2$$
$$\|x\|_2 \le \|x\|_1$$
$$\|x\|_1 \le n \|x\|_\infty$$
$$\|x\|_2 \le \sqrt{n} \|x\|_\infty$$
$$\|x\|_1 \le \sqrt{n} \|x\|_2$$

Pour démontrer cette dernière inégalité, il suffit de remarquer que

$$2 x_i x_j \le x_i^2 + x_j^2$$

Soit $\mathcal{M}$ l'anneau des matrices carrées d'ordre n, représentant un opérateur linéaire de E dans E.

$\forall A \in \mathcal{M}$

$$A = \begin{bmatrix} a_{11} & & a_{1n} \\ & & \\ a_{n1} & & a_{nn} \end{bmatrix} \qquad a_{ij} \in \mathbb{R}$$

On définit une norme induite pour A par

$$\|A\| = \sup \left\{ \frac{\|A(x)\|}{\|x\|} \mid x \in E, x \neq 0 \right\}$$

D'où

$$\|A\|_1 = \max \left\{ \sum_{i=1}^{n} |a_{ij}| \mid 1 \le j \le n \right\}$$ correspondant à $\|x\|_1$

$$\|A\|_2 = \left[\rho(A^T A)\right]^{1/2}$$ ($\rho(A^T A)$ plus grande valeur propre de $A^T A$) correspondant à $\|x\|_2$

$$\|A\|_\infty = \max \left\{ \sum_{j=1}^{n} |a_{ij}| \mid 1 \le i \le n \right\}$$ correspondant à $\|x\|_\infty$

Ces normes vérifient les propriétés suivantes :

$$\|A+B\| \leq \|A\| + \|B\|$$

$$\|\lambda A\| = |\lambda| \, \|A\|$$

$$\|A\| > 0 \iff A \neq 0$$

$$\|AB\| \leq \|A\| . \|B\|$$

$$\|0\| = 0$$

Remarquons aussi que

$$\forall A \in \mathcal{M} \quad \|A\|_2 = \sup_{x \in E-\{0\}} \frac{\|A(x)\|_2}{\|x\|_2}$$

$$= \sup_{x \in E-\{0\}} \frac{\|A(x)\|_2}{\|A(x)\|_1} \times \frac{\|A(x)\|_1}{\|x\|_1} \times \frac{\|x\|_1}{\|x\|_2}$$

en supposant $\|A(x)\|_1 \neq 0$

En tenant compte des inégalités précédentes, on obtient

$$\|A\|_2 \leq \sqrt{n} \, \|A\|_1$$

de même on obtient :

$$\|A\|_2 \leq \sqrt{n} \, \|A\|_\infty \qquad \|A\|_1 \leq n \, \|A\|_\infty \qquad \|A\|_\infty \leq n \, \|A\|_1$$

$$\|A\|_1 \leq \sqrt{n} \, \|A\|_2 \qquad \|A\|_\infty \leq \sqrt{n} \, \|A\|_2$$

De toutes façons,

$$\forall A \in \mathcal{M}, \forall x \in E \qquad \|A(x)\| \leq \|A\| . \|x\|$$

Si x est un vecteur propre correspondant à la valeur propre λ de A, quelle que soit la norme utilisée ,

$$A(x) = \lambda x$$

$$\Rightarrow \quad |\lambda| \, \|x\| \leq \|A\| \, \|x\|$$

$$|\lambda| \leq \|A\|$$

Notons cond (A) = $\|A\| \times \|A^{-1}\|$ le conditionnement de A

cond (A) ≥ 1

Avec la norme $\| \ \|_2$, dans le cas d'une matrice orthogonale, cond(A)=1.

Soit $\rho(A)$ le rayon spectral de A, i.e le nombre positif tel que

$\rho(A) = \max \{ |\lambda_i(A)| \mid 1 \leq i \leq n \}$, les $\lambda_i(A)$ étant les valeurs propres de A.

Quelle que soit la norme, $\rho(A) \leq \|A\|$

Si A est symétrique, $A^T = A$ et $\rho(A) = \|A\|_2$

De même, si A est diagonalisable avec une base de vecteurs propres orthogonaux,

$A = O^{-1} D O$ D diagonale = $\begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix}$

et $O^{-1} = O^T$

$$A^TA = O^TDO\,O^TDO$$

$$= O^TD^2O$$

$$\rho(A) = \|A\|_2$$

On a aussi $\|O\|_2 = 1$

b) Matrice carrée, produit de 2 matrices rectangulaires

Soient E et F deux espaces vectoriels sur $\mathbb{R}$ de dimension respective n et p.

Soient f et g deux applications respectivement de E dans F et de F dans E.

$f : E \to F$

$g : F \to E$

à f correspond une matrice rectangulaire n×p notée C, à g une matrice p×n notée B

Si A représente gof,

$A = B \times C$

$$\|A\| = \sup_{x\in E-\{0\}} \frac{\|A(x)\|}{\|x\|} = \sup\left\{ \frac{\|A(x)\|}{\|B(x)\|} \times \frac{\|B(x)\|}{\|x\|} \;\middle|\; x\in E-\{0\}, B(x)\neq 0 \right\}$$

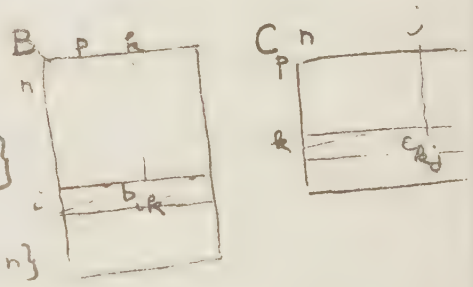
En particulier, considérons les normes $\|\ \|_1$, $\|\ \|_\infty$

$$\|A\|_\infty = \max\left\{ \sum_{j=1}^{n} |a_{ij}| \;\middle|\; 1\le i\le n \right\}$$

$$= \max\left\{ \sum_{j=1}^{n}\sum_{k=1}^{p} |b_{ik}||c_{kj}| \;\middle|\; 1\le i\le n \right\}$$

$$= \max\left\{ \sum_{k=1}^{p} |b_{ik}| \left(\sum_{j=1}^{n} |c_{kj}| \right) \;\middle|\; 1\le i\le n \right\}$$

$$\le \max\left\{ \sum_{j=1}^{n} |c_{kj}| \;\middle|\; 1\le k\le p \right\} \times \max\left\{ \sum_{k=1}^{p} |b_{ik}| \;\middle|\; 1\le i\le n \right\}$$

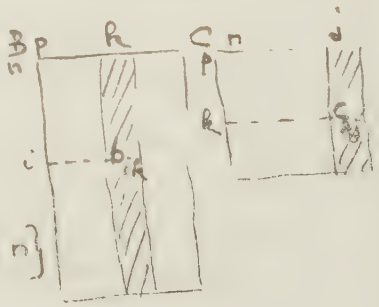


l'égalité ayant lieu notamment lorsque $\sum_{j=1}^{n} |c_{kj}|$ est constant pour tou k.

De même

$$\|A\|_2 = \max\left\{ \sum_{i=1}^{n} |a_{ij}| \;\middle|\; 1\le j\le n \right\}$$

$$\le \max\left\{ \sum_{i=1}^{n} |b_{ik}| \;\middle|\; 1\le k\le p \right\} \max\left\{ \sum_{k=1}^{p} |c_{kj}| \;\middle|\; 1\le j\le n \right\}$$



l'égalité ayant lieu notamment lorsque $\sum_{i=1}^{n} |b_{ik}|$ est constant pour tout k.

c) Etude des variations des valeurs propres

Nous utiliserons une norme matricielle quelconque, vérifiant seulement la propriété suivante :

si D est diagonale alors $\|D\| = \max \{|d_i| \mid 1 \leq i \leq n\}$

d_i étant l'élément diagonal dans la ligne i.

C'est le cas notamment des trois normes considérées précédemment.

Soit une matrice Q diagonisable (nous nous trouvons dans ce cas particulier si Q est semblable à une matrice symétrique,

i.e. $\exists R \in \mathcal{M}_6$ tel que $Q = R^{-1} S R$ avec $S^T = S$).

Considérons $\mathcal{P} = \{P \mid P \in \mathcal{M}_6 \ P^{-1}QP \text{ diagonale}\}$

Comme Q est diagonalisable, $\mathcal{P} \neq \emptyset$

Soit dQ une matrice de perturbation, i.e une matrice quelconque.

Théorème : Les valeurs propres de $Q + dQ$ sont dans la réunion des disques

$$D_i = \{z \in \mathbb{C} \mid |z - \lambda_i| \leq c(Q)\|dQ\|\}$$

avec $c(Q) = \inf\{\text{cond}(P) \mid P \in \mathcal{P}\}$

De plus, s' il existe un entier m vérifiant $1 \leq m < n$ tel que

$$\bigcup_{i=1}^{m} D_i \cap \bigcup_{i=m+1}^{n} D_i = \emptyset \quad , \text{ alors } \bigcup_{i=1}^{m} D_i \text{ contient}$$

exactement m valeurs propres de $Q + dQ$

(on peut toujours supposer qu'il s'agit des m premiers disques).

Soit λ une valeur propre de $Q+dQ$, et soit $P \in \mathcal{P}$

$P^{-1}QP = D$ avec D diagonale

Soient $\lambda_1, \ldots \lambda_i, \ldots \lambda_n$ les valeurs propres de Q

- $\exists i \in [1,n]$, $\lambda = \lambda_i$ le résultat est immédiat
- $\lambda \neq \lambda_i \ \forall i \in [1,n]$; $D - \lambda I$ est régulière, $\exists (D-\lambda I)^{-1} = \begin{bmatrix} \frac{1}{\lambda_1 - \lambda} & & 0 \\ & \ddots & \\ 0 & & \frac{1}{\lambda_n - \lambda} \end{bmatrix}$

$$P^{-1}(Q + dQ - \lambda I)P = P^{-1}QP + P^{-1}dQP - \lambda I$$
$$= D - \lambda I + P^{-1}dQP$$

$D - \lambda I + P^{-1}dQP$ est singulière

et $I + (D-\lambda I)^{-1}P^{-1}dQP$ est singulière .

0 étant valeur propre de $[(D-\lambda I)^{-1}P^{-1}dQP + I]$,

-1 est valeur propre de $(D-\lambda I)^{-1}P^{-1}dQP$;

$$1 \leq \rho\left((D-\lambda I)^{-1}P^{-1}dQP\right) \leq \|(D-\lambda I)^{-1}P^{-1}dQP\|$$

$$\|(D-\lambda I)^{-1}\| = \max\left\{\frac{1}{|\lambda_i-\lambda|} \mid 1\le i\le n\right\} = \frac{1}{\min\{|\lambda_i-\lambda| \mid 1\le i\le n\}}$$

D'où

$$\min\{|\lambda_i-\lambda| \mid 1\le i\le n\} \le \|P\|\cdot\|P^{-1}\|\,\|dQ\|$$

Autrement dit : $\exists i, 1\le i\le n$ tel que

$$|\lambda-\lambda_i| \le \mathrm{cond}(P)\,\|dQ\|$$

, ceci $\forall P\in\mathcal{P}$ donc

$$|\lambda-\lambda_i| \le \varepsilon(P)\,\|dQ\|$$

ceci démontre bien la première partie du théorème.

Supposons maintenant qu'il existe m $(1\le m<n)$ tel que

$$\bigcup_{i=1}^{m} \mathcal{D}_i \cap \bigcup_{i=m+1}^{n} \mathcal{D}_i = \emptyset$$

Considérons la famille de matrices $Q(t) = Q + t\,dQ$, $0\le t\le 1$

Soient $\lambda_i(t)$ les valeurs propres de $Q(t)$. Il est possible de les numéroter pour que chaque fonction $t \mapsto \lambda_i(t)$ soit continue lorsque t décrit l'intervalle $[0,1]$. Pour un indice i donné, l'image du connexe $[0,1]$ par l'application continue $t\mapsto\lambda_i$ est donc une partie connexe du plan complexe. Comme par ailleurs $\lambda_i(0)=\lambda_i$, $\lambda_i(1)=\lambda$, le résultat est immédiat.

En particulier, si $\exists m$ tel que $\mathcal{D}_m \cap \bigcup_{i\neq m}\mathcal{D}_i = \emptyset$

$\mathcal{D}_m$ contient une valeur propre de $Q+dQ$ exactement .

De même, si Q est semblable à une matrice symétrique, i.e

$\exists R\in\mathcal{M}$ tel que $Q = RSR^{-1}$ avec S symétrique ,

Ceci entraîne : $\exists O$ tel que $S = O^T D O$ avec O matrice orthogonale

$$Q = RO^{-1}DOR^{-1}$$

$$\Rightarrow \quad OR\in\mathcal{P}$$

$$\mathrm{cond}(OR) = \|R^{-1}\|\,\|O^{-1}\|\,\|O\|\,\|R\|$$

avec $\|O^{-1}\|_2\,\|O\|_2 = 1$

Ceci exige $|\lambda-\lambda_i| \le \|dQ\|_2\,\|R\|_2\,\|R^{-1}\|_2$.

Soient Q, $Q+dQ$ 2 matrices diagonalisables, à valeurs propres réelles $(\lambda_1\dots\lambda_n)$, $(\lambda'_1\dots\lambda'_n)$

avec $0\le\lambda_n\le\dots\le\lambda_1=1$

$0\le\lambda'_n\le\dots\le\lambda'_1=1$

Soit $\delta = \inf\{1-\lambda_2, \lambda_2-\lambda_3, \dots \lambda_{k-1}-\lambda_k\}$.

Si $\|\delta Q\|_2\|R\|_2\|R^{-1}\|_2 \le \frac{\delta}{2}$, ce théorème conclut aux inégalités suivantes

$$|\lambda_2-\lambda'_2| \le \|dQ\|_2\,\|R\|_2\,\|R^{-1}\|_2$$

$$\vdots$$

$$|\lambda_k-\lambda'_k| \le \|dQ\|_2\,\|R\|_2\,\|R^{-1}\|_2$$

d - Etudes des variations des vecteurs propres - Majoration de l'angle

Pour simplifier l'écriture, nous noterons dans ce paragraphe $\| \ \| = \| \ \|_2$

Soient F, G deux sous espaces vectoriels de E de dimension 1.

Soient a,b deux vecteurs de E engendrant respectivement F,G.

Soient $\tilde{P}$, $\tilde{R}$ les projections orthogonales sur F, G.

L'angle θ des 2 sous espaces est alors déterminée par :

$$\cos\theta = \frac{|a^T b|}{\|a\| \|b\|} \quad , \theta \in [0, \tfrac{\pi}{2}]$$

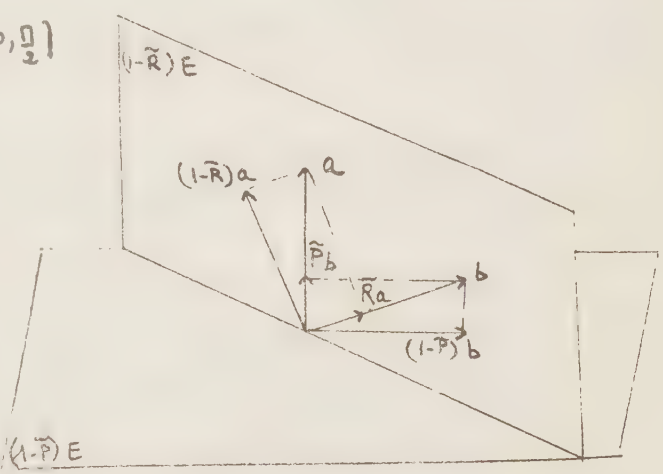
ou en utilisant les projections

$$\sin\theta = \frac{\|\tilde{R}a - a\|}{\|a\|} = \frac{\|\tilde{P}b - b\|}{\|b\|}$$

$$\sin\theta = \sup\left\{ \frac{\|\tilde{P}x - x\|}{\|x\|} \,|\, x \in G - \{0\} \right\}$$

$$= \sup\left\{ \frac{\|\tilde{R}x - x\|}{\|x\|} \,|\, x \in F - \{0\} \right\}$$

$$= \sup\left\{ \frac{\|\tilde{P}\tilde{R}x - \tilde{R}x\|}{\|x\|} \,|\, x \in E - \{0\} \right\}$$



$$\sin\theta = \|\tilde{P}\tilde{R} - \tilde{R}\| = \|(1-\tilde{P})\tilde{R}\| = \|(1-\tilde{R})\tilde{P}\|$$

en appelant $(1-\tilde{P})$ (resp $(1-\tilde{R})$) la projection orthogonale sur le sous espace orthogonal à F (resp G).

Soient $Q, Q+dQ$ deux matrices diagonalisables de $\mathcal{M}$, d'ordre n, (i.e. représentant deux endomorphismes de E).

Soit F (respectivement G) un sous-espace invariant de dimension 1 de Q (respectivement $Q+dQ$). Notons de même $\tilde{P}$ et $\tilde{R}$ les projections orthogonales sur F et G, F' et G' les sous espaces propres complémentaires de F et G. (F et F' ne sont pas forcément orthogonaux, de même G et G')

Q étant diagonalisable, il existe une base de vecteurs propres,

$$E = F + F'$$

$$E = F \oplus F^{\perp} \qquad F^{\perp} \text{ sous espace orthogonal à F.}$$

Notons de même $E = G + G' = G \oplus G^{\perp}$

soit $y \in E$, $\quad y = x + x^{\perp}$ avec $x \in F \quad x^{\perp} \in F^{\perp}$

$x^{\perp} = \tilde{x} + x' \qquad \tilde{x} \in F \quad x' \in F'$

$$(1-\tilde{P})y = x^{\perp} = \tilde{x} + x'$$
$$Q(1-\tilde{P})y = Q\tilde{x} + Qx'$$

D'où $\quad (1-\tilde{P})Q(1-\tilde{P})y = (1-\tilde{P})Qx'$

$$Qy = Q(x + \tilde{x} + x') = Qx + Q\tilde{x} + Qx'$$
$$(1-\tilde{P})Qy = (1-\tilde{P})Qx'$$

Ceci ayant lieu pour tout vecteur de E,

$$(1-\tilde{P})Q(1-\tilde{P}) = (1-\tilde{P})Q$$

F est stable pour Q, $Q\tilde{P} = \tilde{P}Q\tilde{P}$

De façons identiques :

$(1-\tilde{R})\ (Q+dQ)(1-\tilde{R}) = (1-\tilde{R})(Q+dQ)$

$\tilde{R}(Q+dQ)\tilde{R} = (Q+dQ)\tilde{R}$

Etudions la matrice de perturbation dQ :

$$dQ = (Q+dQ) - Q$$

$(1-\tilde{R})dQ\tilde{P} = (1-\tilde{R})(Q+dQ)\tilde{P} - (1-\tilde{R})Q\tilde{P}$

En tenant compte des relations précédentes :

$(1-\tilde{R})dQ\tilde{P} = (1-\tilde{R})(Q+dQ)(1-\tilde{R})\tilde{P} - (1-\tilde{R})\tilde{P}Q\tilde{P}$

Or $(1-\tilde{R})(1-\tilde{R}) = 1-\tilde{R}$

$\tilde{P}\,\tilde{P} = \tilde{P}$

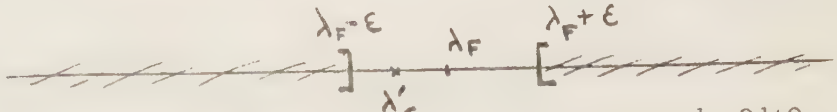
$(1-\tilde{R})dQ\tilde{P} = (1-\tilde{R})\ (Q+dQ)(1-\tilde{R})(1-\tilde{R})\tilde{P} - (1-\tilde{R})\tilde{P}.\tilde{P}Q\tilde{P}$

Notons de même $\mathcal{S}(Q) = \{ M \mid M \in \mathcal{M}_0 \ M^{-1}QM \text{ diagonale} \}$

$z(Q) = \inf \{ \text{cond}(M) \mid M \in \mathcal{S}(Q) \}$

Théorème

Soient Q, Q + dQ deux endomorphismes de E tels que les matrices les représentant soient diagonalisables. Soit $\tilde{P}$ (respectivement $\tilde{R}$) la projection orthogonale sur un sous espace invariant F (respect.G) de dimension 1 de Q (respectivement Q + dQ). Soit λ_F la valeur propre de Q correspondant à F, λ'_G celle correspondant à G. S'il existe un nombre $\varepsilon > 0$ tel que le spectre de la restriction de Q + dQ à G' (sous espace propre de G correspondant aux valeurs propres différentes de λ'_G) soit hors de $[\lambda_F - \varepsilon, \lambda_F + \varepsilon]$ alors $\| (1-\tilde{R})\tilde{P} \| \leq \frac{\| dQ \|}{\varepsilon} z(Q)$



spectre de Q+dQ restreint à G'

spectre de Qd+Q restreint à G'

Ajoutant à Q, Q+dQ un même multiple de l'identité $(-\lambda_F I)$ on se ramène au cas $\lambda_F = 0$. Ceci ne fait que translater les spectres sans changer les vecteurs propres.

D'après la relation précédente, on obtient

$$(1-\tilde{P}).dQ\tilde{P} = (1-\tilde{R})(Q+dQ)(1-\tilde{R})(1-\tilde{R})\tilde{P} - (1-\tilde{R})\tilde{P}.\tilde{P}Q\tilde{P}$$

$$= (1-\tilde{R})(Q+dQ)(1-\tilde{R})(1-\tilde{R})\tilde{P}$$

puisque $\lambda_F = 0$

On a les relations

$$E = G \oplus G^{\perp}$$

$$E = G + G'$$

$\forall\, y^{\perp} \in G^{\perp}$ $\quad y^{\perp} = \beta b + y'$ $\quad \beta \in \mathbb{R}$, $y' \in G'$

$$(1-\tilde{R})(Q+dQ)(1-\tilde{R})\, y^{\perp} = (Q+dQ)\, y'$$

$G \cap G^{\perp} = \{0\} \Rightarrow$

$$\| y' \|^2 = \| y^{\perp} \|^2 + \| \beta b \|^2$$

$$\frac{\| (1-\tilde{R})(Q+dQ)(1-\tilde{R})\, y^{\perp} \|}{\| y^{\perp} \|} = \frac{\| (Q+dQ)\, y' \|}{\| y' \|} \times \frac{\| y' \|}{\| y^{\perp} \|}$$

et $\quad \| y' \| \geq \| y^{\perp} \|$

Soit $M \in \mathcal{P}(Q+dQ)$

$$\forall y' \in G',\ y' \neq 0 \qquad \frac{\|y'\|}{\|(Q+dQ)y'\|} = \frac{\|y'\|}{\|My'\|} \frac{\|My'\|}{\|M(Q+dQ)M^{-1}(My')\|} \times \frac{\|M(Q+dQ)y'\|}{\|(Q+dQ)y'\|}$$

G' est stable par $Q+dQ$, et les valeurs propres de $Q + dQ$ sont en dehors de $[-\varepsilon, +\varepsilon]$, MG' est stable par $M(Q+dQ)M^{-1} = S$ et les valeurs propres sont les mêmes .

$$\sup_{\substack{y' \in G \\ y' \neq 0}} \frac{\|My'\|}{\|M(Q+dQ)M^{-1}(My')\|} < \frac{1}{\varepsilon}$$

$$\sup_{\substack{y' \in G' \\ y' \neq 0}} \frac{\|y'\|}{\|(Q+dQ)y'\|} < \frac{\|M\|\,\|M^{-1}\|}{\varepsilon} \qquad \text{ceci } \forall M \in S(Q+dQ)$$

$$\inf_{\substack{y' \in G \\ y' \neq 0}} \frac{\|(Q+dQ)y'\|}{\|y'\|} > \frac{\varepsilon}{\tau(Q+dQ)}$$

On obtient donc

$$\|(1-\tilde{R})\,dQ\,\tilde{P}\| = \sup_{\substack{x \in E \\ x \neq 0}} \frac{\|(1-\tilde{R})(Q+dQ)(1-\tilde{R})(1-\tilde{R})\tilde{P}x\|}{\|x\|}$$

$$= \sup_{\substack{x \in E \\ x \neq 0}} \frac{\|(1-\tilde{R})(Q+dQ)(1-\tilde{R})(1-\tilde{R})\tilde{P}x\|}{\|(1-\tilde{R})\tilde{P}x\|} \times \frac{\|(1-\tilde{R})\tilde{P}x\|}{\|x\|}$$

En tenant compte des inégalités précédentes

$$\forall y^{\perp} \in G^{\perp} \qquad \frac{\|(1-\tilde{R})(Q+dQ)(1-\tilde{R})\,y^{\perp}\|}{\|y^{\perp}\|} \geqslant \frac{\varepsilon}{\tau(Q+dQ)}$$

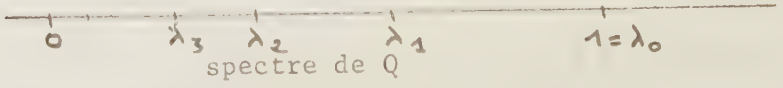
$$\|(1-\tilde{R})\,dQ\,\tilde{P}\| \geqslant \frac{\varepsilon}{\tau(Q+dQ)} \sup_{\substack{x \in E \\ x \neq 0}} \frac{\|(1-\tilde{R})\tilde{P}x\|}{\|x\|}$$

De plus $\|(1-\tilde{R})\| = \|\tilde{P}\| = 1$

$$\|1-\tilde{R})\tilde{P}\| \leq \frac{\|dQ\|\tau(Q+dQ)}{\varepsilon}$$

Le même raisonnement aurait pu se faire avec $Q' = Q+dQ$, $dQ' = -dQ$, d'où le résultat du théorème.

Plaçons-nous dans le cas d'une analyse factorielle des correspondances : Q, $Q+dQ$ sont des matrices diagonalisables à valeurs propres comprises entre 0 et 1 $\quad (1 = \lambda_0 \geqslant \lambda_1 \geqslant \cdots \geqslant \lambda_{n-1} \geqslant 0$ et $1 = \lambda'_0 \geqslant \lambda'_1 \geqslant \cdots \geqslant \lambda'_{n-1} \geqslant 0)$



spectre de Q

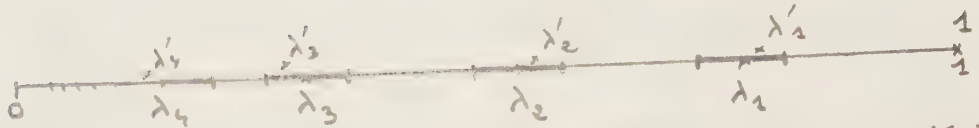
Notons $\quad \delta_k = \inf\,(\lambda_{k+1} - \lambda_k\ ,\ \lambda_k - \lambda_{k-1})$

Supposons $\quad \|dQ\|\,\|R\|\,\|R^{-1}\| \leq \frac{\delta_k}{2}$

les boules $\mathcal{D}_1, \dots \mathcal{D}_k, \bigcup_{i=k+1}^{n} \mathcal{D}_i$ sont disjointes et d'après e,

$$\forall i \in [1,k] \qquad |\lambda_i - \lambda'_i| \leq \|dQ\| \, \|R\| \, \|R^{-1}\|$$

$$\forall i \in [k+1,n] \qquad \lambda'_i \in \bigcup_{i=k+1}^{n} \mathcal{D}_i$$



Dans ce cas, appelant θ_k l'angle de F_k, sous espace propre associé à λ_k et de G_k, sous espace propre associé à λ'_k, nous avons $\sin\theta_k \leq \dfrac{\|dQ\| \, \|R\| \, \|R^{-1}\|}{\delta_k - 2\|dQ\| \, \|R\| \, \|R^{-1}\|}$

e) majoration de la norme de la différence de deux vecteurs

Soient a,b deux vecteurs de E faisant un angle θ entre eux

$$\cos\theta = \frac{\langle a,b \rangle}{\|a\| \, \|b\|}$$

$$\|a-b\|^2 = \|a\|^2 + \|b\|^2 - 2\langle a,b \rangle$$

supposons $\|a\|^2 = \|b\|^2 = 1$

$$\|a-b\|^2 = 2(1-\cos\theta)$$

avec $$\sin\theta \leq \frac{\|dQ\| \, \|R\| \, \|R^{-1}\|}{\delta - 2\|dQ\| \, \|R\| \cdot \|R^{-1}\|}$$

Soit $$x = \frac{\|dQ\| \|R\| \|R^{-1}\|}{\delta}$$

$$\|a-b\| = \sqrt{2 - 2\sqrt{1 - \frac{x^2}{(1-2x)^2}}}$$

$0 \leq x < 1$,

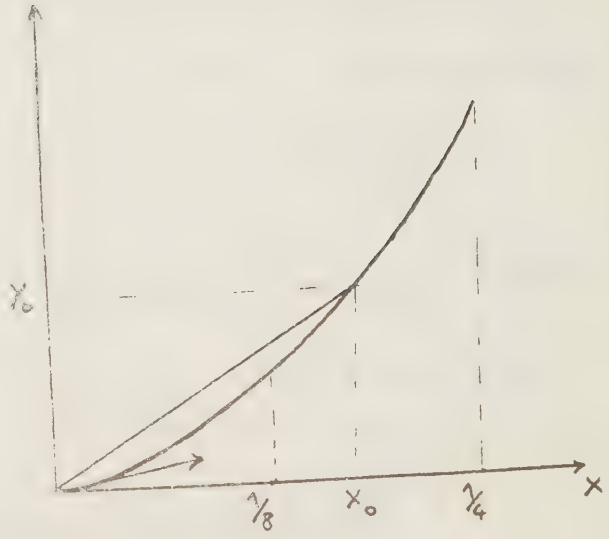
Etudions la fonction $$y = \sqrt{2 - 2\sqrt{1 - \frac{x^2}{(1-2x)^2}}}$$

$$y = \frac{\sqrt{1-x} - \sqrt{1-3x}}{\sqrt{1-2x}}$$

$$y' = \frac{1}{2}\left(\frac{1}{\sqrt{1-x}} - \frac{1}{\sqrt{1-3x}}\right)\frac{1}{(1-2x)^{3/2}}$$

$y'' > 0$

x	0		$\frac{1}{8}$		$\frac{1}{4}$
y'	1	$+$		$+$	
y	0	↗		↗	



$y \leq \frac{y_0}{x_0} x$ pour $x \leq x_0$

Supposons

- $x \leq \frac{1}{4}$

$$\frac{y_0}{x_0} = \frac{\sqrt{3} - 1}{\sqrt{2}} \times 4$$

$$\frac{y_0}{x_0} = 2\sqrt{2}(\sqrt{3} - 1)$$

- $x \leq \frac{1}{8}$

$$\frac{y_0}{x_0} = \frac{2}{6}(\sqrt{7} - \sqrt{5})\sqrt{6}$$

$$\frac{y_0}{x_0} = 2\left(\sqrt{\tfrac{7}{6}} - \sqrt{\tfrac{5}{6}}\right)$$

Nous obtenons donc le résultat suivant

si $\|dQ\| \|R\| \|R^{-1}\| \leq \frac{\sqrt{3}}{4}$ $\quad \|a-b\| \leq 2\sqrt{2}\,(\sqrt{3}-1)\,\frac{\|dQ\| \|R\| \|R^{-1}\|}{\sigma}$

si $\|dQ\| \|R\| \|R^{-1}\| \leq \frac{\sigma}{8}$ $\quad \|a-b\| \leq 8(\sqrt{7/6}-\sqrt{5/6})\,\frac{\|dQ\| \|R\| \|R^{-1}\|}{\sigma}$

Supposons maintenant que les vecteurs considérés V_1, V_2 ne soient plus normés mais soient de la forme $V_1 = \alpha a$, $V_2 = \beta b$

avec a et b normés

$$\alpha = \frac{1}{\|T_1 a\|} \quad , \beta = \frac{1}{\|T_2 b\|}$$

T_1 et T_2 étant 2 endophormismes de E

$$\alpha \leq \|T_1^{-1}\| \qquad \beta \leq \|T_2^{-1}\|$$

$$\|V_1 - V_2\| = \|\alpha a - \beta b\| \leq \alpha \|a-b\| + |\alpha-\beta|$$

$$|\alpha-\beta| = \left| \frac{1}{\|T_1 a\|} - \frac{1}{\|T_2 b\|} \right| \leq \frac{1}{\|T_1 a\| \|T_2 b\|} \left[\|T_1 a - T_2 b\| \right]$$

$$\|T_1 a - T_2 b\| \leq \|T_1 (a-b)\| + \|(T_1 - T_2) b\|$$

$$\|V_1 - V_2\| \leq \|T_2^{-1}\| \|a-b\| + \|T_1^{-1}\| \|T_2^{-1}\| \left\{ \|T_1\| \|a-b\| + \|T_1 - T_2\| \right\}$$

$$\leq \|a-b\| \|T_2^{-1}\| (1 + \|T_2^{-1}\| \|T_1\|) + \|T_1^{-1}\| \|T_2^{-1}\| \|T_1 - T_2\|$$

On obtient alors les relations

si $\|dQ\| \|R\| \|R^{-1}\| \leq \frac{\sigma}{8}$

$$\|V_1 - V_2\| \leq 8(\sqrt{7/6} - \sqrt{5/6}) \frac{\|dQ\|}{\sigma} \|R\| \|R^{-1}\| \|T_1^{-1}\| (1 + \|T_2^{-1}\| \|T_1\|) + \|T_2^{-1}\| \|T_2^{-1}\| \|T_1 - T_2\|$$

2- Etude des variations de Q en fonction des "trous"

Les matrices Q sont définies par

$Q = BC$

$$Q = [q_{jj'}] \qquad q_{jj'} = \sum_{i \in I} \frac{k_{ij}}{k_j} \frac{k_{ij'}}{k_i}$$

$$B = \overset{J \backslash I}{j} \begin{pmatrix} & & i \\ & & \vdots \\ \cdots & & \frac{k_{ij}}{k_j} \end{pmatrix} \qquad C = \overset{I \backslash J}{i} \begin{pmatrix} & & j' \\ & & \vdots \\ \cdots & & \frac{k_{ij'}}{k_i} \end{pmatrix}$$

Q est semblable à une matrice symétrique S

$$S = R Q R^{-1}$$

avec $R = \begin{bmatrix} \ddots & & 0 \\ & \sqrt{k_j} & \\ 0 & & \ddots \end{bmatrix}$ $\qquad R^{-1} = \begin{bmatrix} \ddots & & 0 \\ & \frac{1}{\sqrt{k_j}} & \\ 0 & & \ddots \end{bmatrix}$

les vecteurs propres V vérifient

$$\| TV \|_2 = 1 \quad \text{avec} \quad T = \frac{R}{\sqrt{k}}$$

Si $Q_1 = B_1 C_1$, $Q_2 = B_2 C_1$, cela entraîne

$$Q_1 - Q_2 = B_1 (C_1 - C_2) + (B_1 - B_2) C_2$$

a) Un seul trou en (i_o, j_o)

Notons $a = k_{i_o} = \Sigma \{ k_{ij} \mid j \in J - \{j_o\} \}$ $\qquad b = k_{j_o} = \Sigma \{ k_{ij_o} \mid i \in I - \{i_o\} \}$

Considérons deux valeurs successives, pour l'une $k_{i_o j_o} = X_1$

pour l'autre $k_{i_o j_o} = X_2$

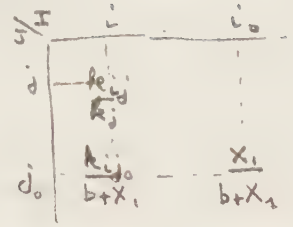
Notons $\Delta X = X_2 - X_1$

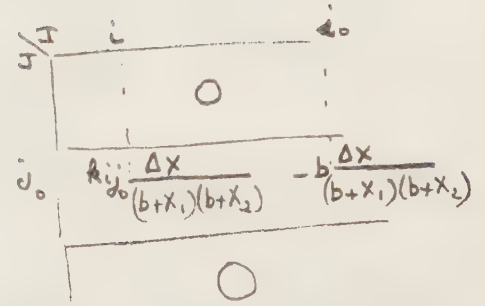
$\Delta Q = Q_1 - Q_2$

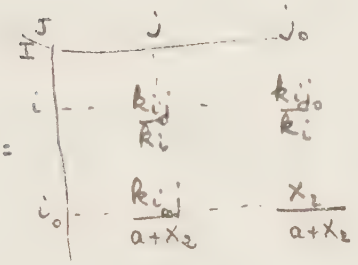
$\Delta B = B_1 - B_2$

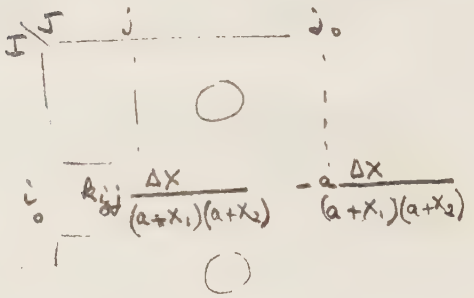
$\Delta C = C_1 - C_2$

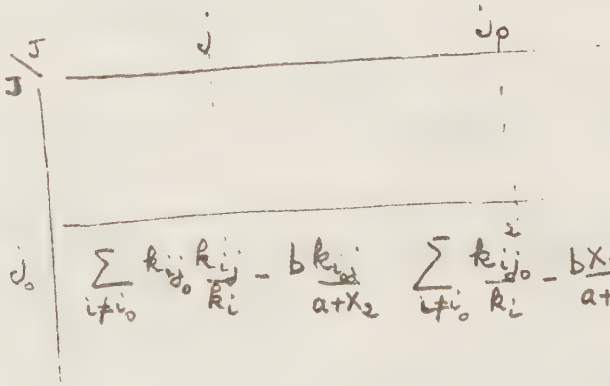
Nous avons

$B_1 =$ 

$\Delta B :$ 

$C_2 =$ 

$\Delta C :$ 

$\frac{(b+X_1)(b+X_2)}{\Delta X}\,\Delta B C_2 =$ 

pour calculer $\|\Delta B C_2\|_2$, il nous faut calculer une expression de la forme

$$\sup_{x \in E} \frac{|\delta_1 x_1 + \cdots + \delta_n x_n|}{(x_1^2 + \cdots + x_n^2)^{1/2}}$$

avec $\delta_1 + \cdots + \delta_n = 0$

or $$\sup_{x \in E} \frac{|\delta_1 x_1 + \cdots + \delta_n x_n|}{(x_1^2 + \cdots + x_n^2)^{1/2}} \leq \sup_{x \in E} \left\{ \max_\ell \delta_\ell \times \frac{\left(\frac{|\delta_1|}{|\delta_\ell|}|x_1| + \cdots + \frac{|\delta_n|}{|\delta_\ell|}|x_n|\right)}{(x_1^2 + \cdots + x_n^2)^{1/2}} \right.$$

$$\leq \max_\ell \delta_\ell \times \sup_{x \in E} \frac{\Sigma |x_i|}{(\Sigma x_i^2)^{1/2}}$$

$$\leq \sqrt{p}\, \max_\ell \delta_\ell$$

si p est le cardinal de J

$$\|\Delta B C_2\|_2 \leq \frac{\sqrt{p}\,|\Delta x|}{(b+x_1)(b+x_2)} \max_{j\in J}\left\{\left|\sum_{i\neq i_0} k_{ij_0}\frac{k_{ij}}{k_i} - b\,\frac{k_{i_0 j}}{k_{i_0}}\right|\right\}$$

on se trouve dans le cas suivant

$$\sum \alpha_j = \sum \beta_j = b \qquad \alpha_j, \beta_j > 0$$

et on cherche $\max_{j\in J} |\alpha_j - \beta_j| \leq \max\left\{\max_{j\in J}\alpha_j - \min_{j\in J}\beta_j\,;\, \max_{j\in J}\beta_j - \min_{j\in J}\alpha_j\right\}$

$$\|\Delta B C_2\|_2 \leq \frac{\sqrt{p}\,|\Delta x|}{(b+x_1)(b+x_2)} \max\left\{\max_{j\in J}\left\{\sum_{i\neq i_0} k_{ij_0}\frac{k_{ij}}{k_i}\right\} - b\min_{j\in J}\frac{k_{i_0 j}}{k_{i_0}}\,;\, b\max_{j\in J}\frac{k_{i_0 j}}{k_{i_0}} - \min_{j\in J}\left\{\sum_{i\neq i_0} k_{ij_0}\frac{k_{ij}}{k_i}\right\}\right.$$

De même

$$\frac{(a+x_1)(a+x_2)}{\Delta x} B_1 \Delta C =$$

J \ J	… j' …	… j_0
j	… $\frac{k_{i_0 j} k_{i_0 j'}}{k_j}$ …	… $\frac{k_{i_0 j} a}{k_j}$
j_0	… $\frac{k_{i_0 j_0} k_{i_0 j'}}{k_{j_0}}$ …	… $\frac{k_{i_0 j_0} a}{k_{j_0}}$

Etudions $\dfrac{\|\Delta C(x)\|_2}{\|x\|_2}$

$$\frac{(a+x_1)(a+x_2)\,\Delta C(x)}{\Delta x} = \begin{vmatrix} 0 \\ \vdots \\ 0 \\ \sum_{j\neq j_0} k_{i_0 j} x_j - a x_{j_0} \\ 0 \\ \vdots \\ 0 \end{vmatrix} \quad (i_0)$$

nous sommes donc amenés à étudier une expression de la forme

$$\frac{\left(x_{j_0} + \sum_{j\neq j_0} \alpha_j x_j\right)^2}{\sum_{j\in J} x_j^2} \qquad \text{avec} \qquad \alpha_j > 0 \quad \sum_{j\neq j_0}\alpha_j = 1 \;,\; x_j \geq 0$$

or

$$\left(x_{j_0} + \sum_{j\neq j_0}\alpha_j x_j\right)^2 \leq \left(x_{j_0} + \max_{j\neq j_0} x_j\right)^2$$

$$\Rightarrow \quad \frac{\|\Delta C(x)\|_2}{\|x\|_2} \leq \frac{\sqrt{2}\,|\Delta x|\,a}{(a+x_1)(a+x_2)}$$

Etudions $\dfrac{\|B_1(x)\|_2}{\|x\|_2}$ avec x de la forme $\begin{vmatrix} 0 \\ \vdots \\ x_{i_0} \\ 0 \\ \vdots \end{vmatrix}$

$$B_1(x): \begin{vmatrix} \frac{k_{i_0 j}}{k_j} x_{i_0} \\ \vdots \end{vmatrix} \quad \Rightarrow \quad \frac{\|B_1(x)\|_2}{\|x\|_2} = \left\{\sum_{j\in J}\left(\frac{k_{i_0 j}}{k_j}\right)^2\right\}^{1/2}$$

en particulier $\dfrac{\|B_1(x)\|_2}{\|x\|_2} \leq \sum_{j\in J}\dfrac{k_{i_0 j}}{k_j}$ ou $\dfrac{\|B_1(x)\|_2}{\|x\|_2} \leq \sqrt{p}\,\max_{j\in J}\dfrac{k_{i_0 j}}{k_j}$

avec p = card J.

D'où $\|B_1 \Delta C\|_2 = \sup_{x \in E} \frac{\|B_1 \Delta C x\|_2}{\|x\|_2} \leq \sup_{x \in E} \frac{\|B_1 \Delta C x\|_2}{\|\Delta C x\|_2} \sup_{x \in E} \frac{\|\Delta C x\|_2}{\|x\|_2}$

$$\leq \frac{\sqrt{2}\,|\Delta x|\,a}{(a+x_1)(a+x_2)} \left[\sum_{j \in J} \left(\frac{k_{i_0 j}}{k_j}\right)^2\right]^{1/2}$$

On obtient

$$\|\Delta Q\|_2 \leq \frac{\sqrt{p}\,|\Delta x|}{(b+x_1)(b+x_2)} \max_{j \in J} \left\{ \left| \sum_{i \neq i_0} \frac{k_{i j_0}}{k_i} k_{ij} - b\, k_{i_0 j} \right| \right\} + \frac{\sqrt{2}\,|\Delta x|\,a}{(a+x_1)(a+x_2)} \left[\sum_{j \in J} \left(\frac{k_{i_0 j}}{k_j}\right)^2\right]^{1/2}$$

$$\leq \frac{\sqrt{p}\,|\Delta x|}{b} \max_{j \in J} \left\{ \left| \sum_{i \neq i_0} \frac{k_{i j_0}}{k_i} k_{ij} - b\, k_{i_0 j} \right| \right\} + \frac{\sqrt{2}\,|\Delta x|}{a} \left[\sum_{j \in J} \left(\frac{k_{i_0 j}}{k_j}\right)^2\right]^{1/2}$$

b- N trous dans des colonnes et lignes distinctes

Notons $(i_1, j_1) \cdots (i_N, j_N)$ les indices correspondant aux trous

$\Delta B =$

	I	i_1	i_n
J			
j_1		$-b_1 \frac{\Delta x_1}{(b_1+x_{11})(b_1+x_{12})}$	$k_{i_n j_1} \frac{\Delta x_1}{(b_1+x_{11})(b_1+x_{12})}$
j_n	$k_{i j_n} \frac{\Delta x_n}{(b_n+x_{n1})(b_n+x_{n2})}$		$-b_n \frac{\Delta x_n}{(b_n+x_{n1})(b_n+x_{n2})}$
j_N			

On note de même

$$\Delta x_n = x_{n1} - x_{n2} \qquad \forall n \in [1, N]$$

x_{n1}, x_{n2} étant les 2 valeurs successives du trou (i_n, j_n)

seules sont non nulles les lignes $j_1, \ldots j_n, j_N$

$\Delta B C_2 =$

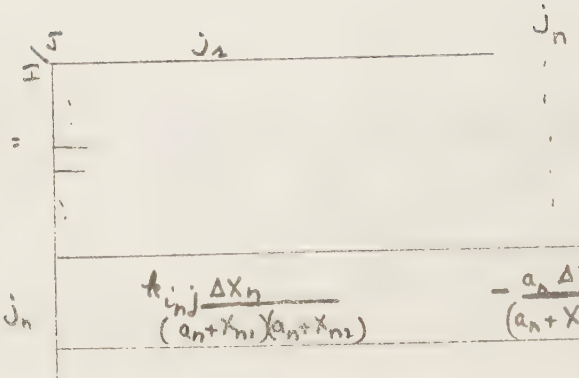
	J	j	j_n
J			
j_n	$\left(\sum_{i \neq i_n} k_{ij_n} \frac{k_{ij}}{k_i} - b_n \frac{k_{i_n j}}{k_{i_n}}\right) \frac{\Delta x_n}{(b_n+x_{n1})(b_n+x_{n2})}$		$\left(\sum_{i \neq i_n} \frac{k_{ij_n}^2}{k_i} - b_n \frac{k_{i_n j_n}}{k_{i_n}}\right) \frac{\Delta x_n}{(b_n+x_{n1})(b_n+x_{n2})}$

de même seules les lignes $j_1, \ldots j_N$ sont non nulles.

Dans ce cas on obtient,

$$\|\Delta B C_2\|_2 \leq \sum_{n \in [1,N]} \frac{\sqrt{p}\,|\Delta x_n|}{(b_n+x_{n1})(b_n+x_{n2})} \max_{j \in J} \left\{ \left| \sum_{i \neq i_n} k_{ij_n} \frac{k_{ij}}{k_i} - b\, \frac{k_{i_n j}}{k_{i_n}} \right| \right\}$$

De même



$$\Rightarrow \frac{\|\Delta C X\|_2}{\|X\|_2} \leq \sum_{n\in[1,N]} \frac{\sqrt{2}\,|\Delta X_n|\,a_n}{(a_n+X_{n1})(a_n+X_{n2})}$$

pour x de la forme $\begin{pmatrix} 0 \\ \vdots \\ X_{i_1} \\ 0 \\ X_{i_n} \\ \vdots \end{pmatrix}$, on obtient

$$\frac{\|B_1(x)\|_2^2}{\|X\|_2^2} = \frac{\sum_{j\in J}\left(\sum_{n\in[1,N]} \frac{k_{i_n j}}{k_j} X_{i_n}\right)^2}{\sum_{n\in[1,N]} X_{i_n}^2} \leq N \sum_{j\in J}\left\{\max_{n\in[1,N]} \frac{k_{i_n j}}{k_j}\right\}^2$$

D'où

$$\|B_1 \Delta C\|_2 \leq \left\{\sum_{n\in[1,N]} \frac{\sqrt{2}\,|\Delta X_n|\,a_n}{(a+X_{n1})(a+X_{n2})}\right\} \times \sqrt{N}\left\{\sum_{j\in J}\left(\max_{n\in[1,N]} \frac{k_{i_n j}}{k_j}\right)^2\right\}^{1/2}$$

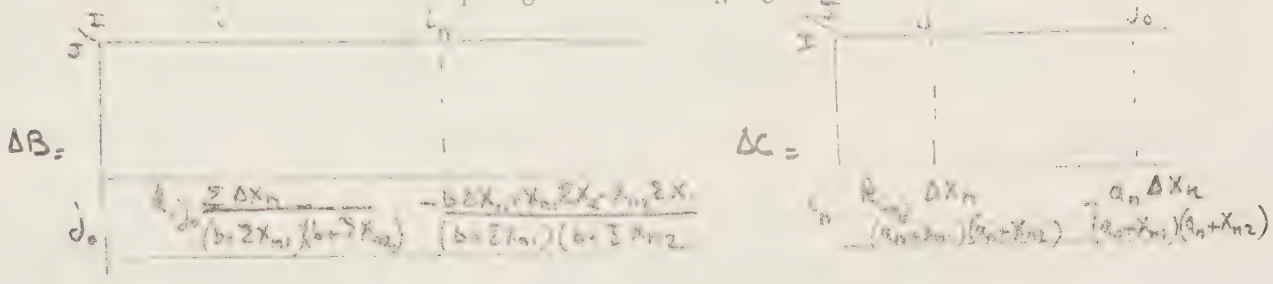
$$\|\Delta Q\|_2 \leq \sum_{n\in[1,N]} \frac{\sqrt{P}\,|\Delta X_n|}{b_n^2} \max_{j\in J}\left\{\left|\sum_{i\neq i_n} k_{i j_n} \frac{k_{ij}}{k_i} - b_n \frac{k_{i_n j}}{k_{i_n}}\right|\right\} +$$

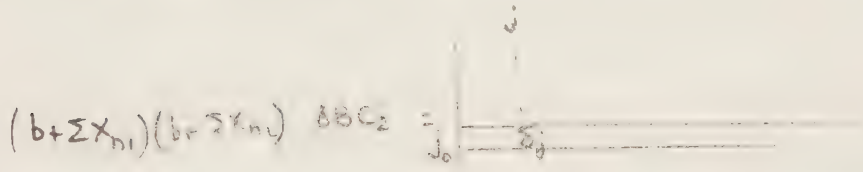
$$+ \sqrt{N}\left[\sum_{j\in J}\left(\max_{n\in[1,N]} \frac{k_{i_n j}}{k_j}\right)^2\right]^{1/2} \times \left\{\sum_{n\in[1,N]} \frac{\sqrt{2}\,|\Delta X_n|}{a_n}\right\}$$

$$\|\Delta Q\|_2 \leq \sum_{n\in[1,N]} |\Delta X_n| \left\{\frac{\sqrt{P}}{b_n^2} \max_{j\in J}\left\{\left|\sum_{i\neq i_n} k_{i j_n} \frac{k_{ij}}{k_i} - b_n \frac{k_{i_n j}}{k_{i_n}}\right|\right\} + \frac{\sqrt{2N}}{a_n}\left[\sum_{j\in J}\left(\max_{n\in[1,N]} \frac{k_{i_n j}}{k_j}\right)^2\right]^{1/2}\right\}$$

c-N Trous dans une même colonne j_o

Nous avons donc N trous en $(i_1, j_o), \ldots\ldots (i_N, j_o)$





$$\sigma_j = \sum_{i \neq i_1 \ldots i_N} k_{ij_o} \frac{k_{ij}}{k_i} (\Sigma \Delta X_n) + \sum_{n=1}^{N} \left\{ -b \frac{k_{i_n j}}{k_{i_n}} \Delta X_n + X_{n1} (\Sigma X_{n2}) \frac{k_{i_n j}}{k_{i_n}} - X_{n2} (\Sigma X_{n1}) \frac{k_{i_n j}}{k_{i_n}} \right\}$$

$$= \sum_{n=1}^{N} \left\{ \sum_{i \neq i_1 \ldots i_N} k_{ij_o} \frac{k_{ij}}{k_i} - b \frac{k_{i_n j}}{k_{i_n}} \right] (\Delta X_n) \Big\} + \left(\sum_{n=1}^{N} X_{n2} \right) \left(\sum_{n=1}^{N} X_{n1} \frac{k_{i_n j}}{k_{i_n}} \right) - \left(\sum_{n=1}^{N} X_{n1} \right) \left(\sum_{n=1}^{N} X_{n2} \frac{k_{i_n j}}{k_{i_n}} \right)$$

$$\left(\sum_{n=1}^{N} X_{n2} \right) \left(\sum_{n=1}^{N} X_{n1} \frac{k_{i_n j}}{k_{i_n}} \right) - \left(\sum_{n=1}^{N} X_{n1} \right) \left(\sum_{n=1}^{N} X_{n2} \frac{k_{i_n j}}{k_{i_n}} \right) = \sum_{n=1}^{N} \Delta X_n \left(\sum_{n'=1}^{N} X_{n'1} \left(\frac{k_{i_{n'} j}}{k_{i_{n'}}} - \frac{k_{i_n j}}{k_{i_n}} \right) \right)$$

d'après b), nous aurons

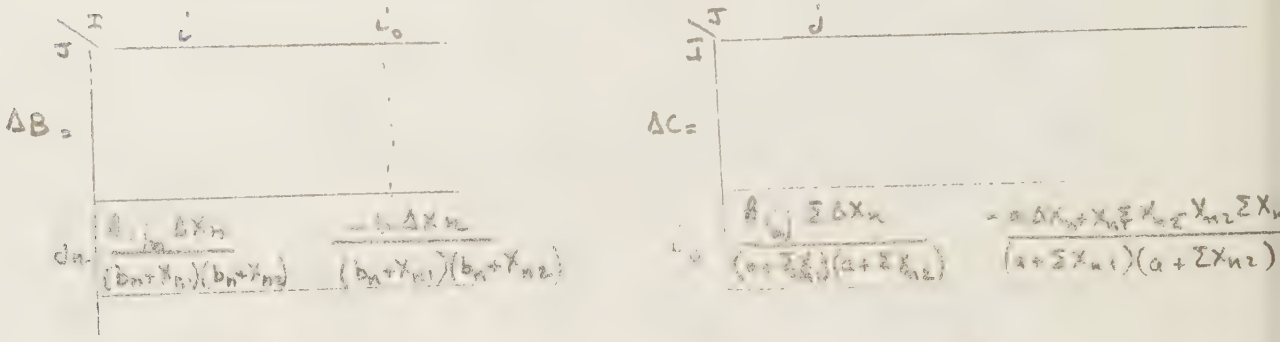
$$\| \Delta B C_2 \|_2 \leq \frac{\Sigma |\Delta X_n| \sqrt{p}}{(b + \Sigma X_{n1})(b + \Sigma X_{n2})} \left\{ \max_{\substack{j \in J \\ n \in [1,N]}} \left\{ \left| \sum_{i \neq i_1 \ldots i_N} k_{ij_o} \frac{k_{ij}}{k_i} - b \frac{k_{i_n j}}{k_{i_n}} \right| \right\} + \max_{\substack{j \in J \\ n' \neq n}} \left\{ \left\| \frac{k_{i_n j}}{k_{i_n}} - \frac{k_{i_{n'} j}}{k_{i_{n'}}} \right| \right\} (\Sigma X_n \right.$$

$$\frac{\| \Delta C(X) \|_2}{\| X \|_2} \leq \sum_{n=1}^{N} \frac{\sqrt{2} |\Delta X_n| a_n}{(a_n + X_{n1})(a_n + X_{n2})} \qquad \frac{\| B_n(X) \|}{\| X \|} \leq \sqrt{N} \left[\sum_{j \in J} \left(\max_{n \in [1,N]} \frac{k_{i_n j}}{k_j} \right)^2 \right]^{1/2}$$

D'où

$$\| B_1 \Delta C \|_2 \leq \sqrt{N} \left[\sum_{j \in J} \left(\max_{n \in [1,N]} \frac{k_{i_n j}}{k_j} \right)^2 \right]^{1/2} \left[\sum_{n=1}^{N} \frac{\sqrt{2} |\Delta X_n|}{a_n} \right]$$

d- Les trous sont dans une même ligne $i_o : (i_o, j_1), \ldots (i_o, j_N)$



$$\|\Delta B C_2\|_2 \leq \sum_{n=1}^{N} \frac{\sqrt{p}\,|\Delta X_n|}{(b_n+X_{n1})(b_n+X_{n2})} \max_{j\in J}\left\{ \left| \sum_{i\neq i_o} k_{ij_n} \frac{k_{ij}}{k_i} - b_n \frac{k_{i_o j}}{k_{i_o}} \right| \right\}$$

$$\|B_1 \Delta C\|_2 \leq \left[\sum_{j\in J} \left(\frac{k_{i_o j}}{k_j} \right)^2 \right]^{1/2} \times \sqrt{N+1}\; \frac{|\Sigma \Delta X_n|\,(a+\Sigma X_{n1})}{(a+\Sigma X_{n1})(a+\Sigma X_{n2})}$$

puisque $$a\,\Delta X_n - X_{n1}(\Sigma X_{n2}) + X_{n2}\Sigma X_{n1} = \Delta X_n (a+\Sigma X_{n1}) - X_{n1}\Sigma(\Delta X_n)$$

$$|\Delta X_n (a+\Sigma X_{n1}) - X_{n1}\Sigma \Delta X_n| \leq (a+\Sigma X_{n1}|\Sigma \Delta X_n|$$

e- Cas général

De la même façon, nous décomposerons ΔQ, avec

$$\|\Delta Q\|_2 \leq \|B_1 \Delta C\|_2 + \|\Delta B\, C_2\|_2$$

Soit $\mathcal{N}$ l'ensemble des N trous.

Soient $i_1, \ldots i_{N_i}$ les i correspondants aux trous

$j_1, \ldots j_{N_j}$ les j correspondants aux trous

$N_i \leq N$, $N_j \leq N$, si $N_i = N_j = N$ les trous sont dans des lignes et des colonnes distinctes (cas b).

$N_j = 1$ (resp. $N_i = 1$) nous sommes dans le cas c (resp. cas d)

Soit $$J_n = \left\{ j_R \mid R \in [1, N_j] \;, (i_n, j_R) \in \mathcal{N} \right\}$$

$$I_n = \{ i_k \mid k \in [1, N_i] \ , (i_k, j_m) \in \mathcal{P} \}$$

Notons $\mathcal{N}_i = \{ k \mid \exists j \in J \quad (i_k, j) \in \mathcal{P} \}$

$\mathcal{N}_j = \{ k \mid \exists i \in I \quad (i, j_k) \in \mathcal{P} \}$

$\forall n \in \mathcal{N}_i \qquad a_n = \Sigma \{ k(i_n, j) \mid j \in J - J_n \}$

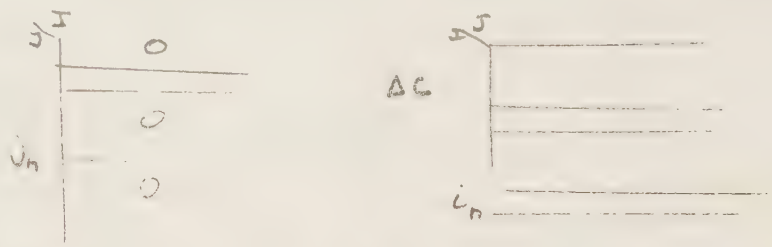
$\forall n \in \mathcal{N}_j \qquad b_n = \Sigma \{ k(i, j_n) \mid i \in I - I_n \}$

$$\Sigma \{ \text{card } J_n \mid n \in \mathcal{N}_i \} = \Sigma \{ \text{card } I_n \mid n \in \mathcal{N}_j \} = N$$

Notons X_{ij1} et X_{ij2} les valeurs successives du trou (i,j) et $\Delta X_{ij} = X_{ij1} - X_{ij2}$

Opérant de la même façon

ΔBC_2 est de la forme



$$\|\Delta BC_2\|_2 \leq \sum_{n \in \mathcal{N}_j} \left[\frac{\sum_{i \in I_n} |\Delta X_{ij_n}| \sqrt{p}}{(b_n + \sum_{i \in I_n} X_{ij_n 1})(b_n + \sum_{i \in I_n} X_{ij_n 2})} \left(\max_{\substack{j \in J \\ n \in I_n}} \left\{ \left| \sum_{i \in I_n} \frac{k_{ij_n} k_{ij}}{k_i} - b \frac{k_{ij_n}}{k_i} \right| \right\} \right) + \right.$$

$$\left. \max_{j \in J} \left\{ \max_{i, i' \in I_n} \left(\left| \frac{k_{ij}}{k_i} - \frac{k_{i'j}}{k_{i'}} \right| \right) \times \left(\sum_{i \in I_n} X_{ij_n 1} \right) \right\} \right]$$

$$\|B_1 \Delta C\|_2 \leq \sqrt{\text{card } I_n} \left\{ \sum_{j \in J} \left(\max_{i \in I_n} \frac{k_{ij}}{k_j} \right)^2 \right\}^{\frac{1}{2}} \cdot \left\{ \sum_{i \in \mathcal{N}_i} \left[\sqrt{\text{card } J_n + 1} \ \frac{\left(\sum_{j \in J_n} |\Delta X_{ij}| \right)(a_n + \sum_{j \in J_n} X_{ij_n})}{(a_n + \sum_{j \in J_n} X_{ij_n 1})(a_n + \sum_{j \in J_n} X_{ij_n 2})} \right] \right\}$$

d'où $\|\Delta Q\|_2 \leq \|\Delta BC_2\|_2 + \|B_1 \Delta C\|_2$

3 - Etude des variations des valeurs des trous en fonction de K

a- Un seul trou

Soit x la valeur trouvée pour K, X et K sont liés par la formule

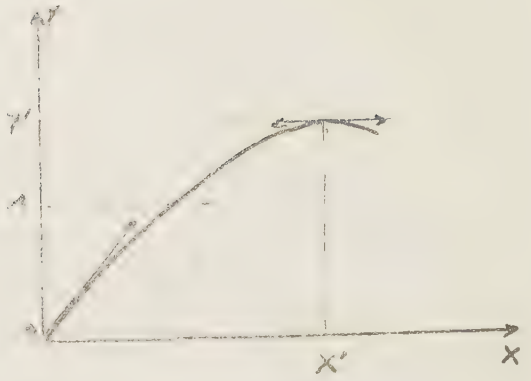
$$\frac{x(c+x)}{(a+x)(b+x)} = K$$

Etudions la fonction $\quad Y = \dfrac{x(c+x)}{(a+x)(b+x)}$, $\dfrac{dY}{dx} = \dfrac{x^2(-c+a+b) + 2abx + abc}{(a+x)^2(b+x)^2}$

$\dfrac{dY}{dx} = 0$ pour $\quad x' = \dfrac{ab}{c-a-b}\left[1 + \sqrt{\dfrac{(c-a)(c-b)}{ab}}\right]$

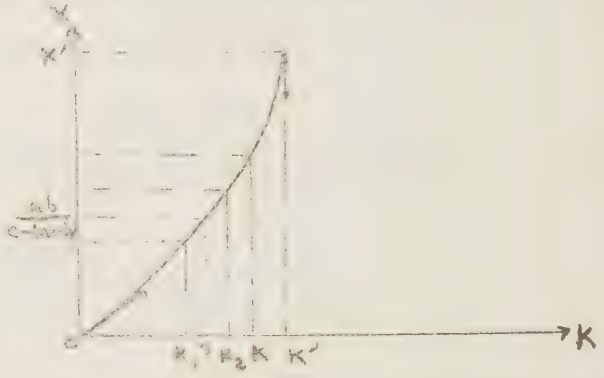
$$Y' = \frac{c^2}{\left[\sqrt{a(c-b)} + \sqrt{b(c-a)}\right]^2}$$

X	0		$\frac{ab}{c-(a+b)}$		X'		∞
$\frac{dY}{dX}$	$\frac{c}{ab}$		$\frac{[c-(a+b)]^2}{ab(c-a)(c-b)}$		0		
Y	0	$\nearrow$	1	$\nearrow$	Y'	$\searrow$	1



Nous obtenons les variations suivantes de X en fonction de K.

K	0		1		K'
$\frac{dX}{dK}$	$\frac{ab}{c^2}$	$+$		$+$	$\parallel$
X	0	$\nearrow$	$\frac{ab}{c-(a+b)}$	$\nearrow$	X'



Soient K_1, K_2

$K_1 < K < K'$

$K_2 < K < K'$

$$\frac{X_1 - X_2}{K_1 - K_2} < \alpha$$

avec $\alpha = \dfrac{(a+X)^2(b+X)^2}{X^2(-c+a+b)+2abx+abc}$ calculé pour la valeur X correspondant à K.

b- N trous dans des lignes et des colonnes distinctes

$\forall n \in [1, N]$

$$K_n = \frac{X_n\left(c + \sum_{k=1}^{N} X_k\right)}{(a_n + X_n)(b_n + X_n)} = F_n(X_1, \dots X_n)$$

$k \neq n$

$$\frac{\partial F_n}{\partial X_k} = \frac{X_n}{(a_n + X_n)(b_n + X_n)}$$

$$\frac{\partial F_n}{\partial x_n} = \frac{x_n}{(a_n+x_n)(b_n+x_n)} + \left(c+\sum_{k=1}^{N} x_k\right)\frac{a_n b_n - x_n^2}{(a_n+x_n)^2(b_n+x_n)^2}$$

Le Jacobien J est donc de la forme :

$$J = \begin{pmatrix} \alpha_1+\beta_1 & \alpha_1 & \cdots & \alpha_1 & \cdots & \alpha_1 \\ \alpha_2 & & & & & \alpha_2 \\ \vdots & & & & & \vdots \\ \alpha_n & \cdots & \alpha_n+\beta_n & \cdots & & \alpha_n \\ \vdots & & & & & \vdots \\ \alpha_N & \alpha_N & \cdots & \cdots & & \alpha_N+\beta_N \end{pmatrix}$$

avec

$$\alpha_n = \frac{x_n}{(a_n+x_n)(b_n+x_n)}$$

$$\beta_n = \left(c+\sum_{k=1}^{N} x_k\right)\frac{a_n b_n - x_n^2}{(a_n+x_n)^2(b_n+x_n)^2}$$

Soit $J^{-1} = \begin{pmatrix} x_{11} & \cdots & x_{1N} \\ \vdots & & \vdots \\ x_{N1} & \cdots & x_{NN} \end{pmatrix}$

pour avoir $J.J^{-1} = I$, cela exige:

$\forall i \in [1,N] \quad \forall k \neq i$

$$\beta_k x_{ki} + \alpha_k \left(\sum_l x_{li}\right) = 0$$

$$\beta_i x_{ii} + \alpha_i \left(\sum_l x_{li}\right) = 1$$

$$x_{ki} = -\frac{\alpha_k}{\beta_k}\left(\sum_l x_{li}\right)$$

$$x_{ii} = \frac{1-\alpha_i\left(\sum_l x_{li}\right)}{\beta_i}$$

$$\sum_l (x_{li}) = \frac{1}{\beta_i} - \sum_k \frac{\alpha_k}{\beta_k}\left(\sum_l x_{li}\right) \qquad \sum_l x_{li} = \frac{1}{\beta_i}\alpha$$

avec $\alpha = \dfrac{1}{1+\sum_k \frac{\alpha_k}{\beta_k}}$ $\qquad x_{ki} = -\dfrac{\alpha_k}{\beta_k \beta_i}\alpha$ $\qquad x_{ii} = \dfrac{\beta_i - \alpha_i \alpha}{\beta_i^2}$

$$J^{-1} = \begin{pmatrix} \frac{1}{\beta_1} - \frac{\alpha_1}{\beta_1}\frac{\alpha}{\beta_1} & \cdots & -\frac{\alpha_1}{\beta_1}\frac{\alpha}{\beta_k} & \cdots & -\frac{\alpha_1}{\beta_1}\frac{\alpha}{\beta_N} \\ \vdots & & & & \\ -\frac{\alpha_k}{\beta_k}\frac{\alpha}{\beta_1} & \cdots & \frac{1}{\beta_k} - \frac{\alpha_k}{\beta_k}\frac{\alpha}{\beta_k} & \cdots & -\frac{\alpha_k}{\beta_k}\frac{\alpha}{\beta_N} \\ \vdots & & & & \\ -\frac{\alpha_N}{\beta_N}\frac{\alpha}{\beta_1} & \cdots & & & \frac{1}{\beta_N} - \frac{\alpha_N}{\beta_N}\frac{\alpha}{\beta_N} \end{pmatrix}$$

Ceci suppose

$$\sum_{k=1}^{N} \frac{\alpha_k}{\beta_n} = \sum_{n=1}^{N} \frac{x_n (a+x_n)(b+x_n)}{(c+\sum_{k=1}^{N} x_k)(a_n b_n - x_n^2)} \neq 1$$

(c'est le cas notamment si $x_n < \sqrt{a_n b_n}$)

$$\| J^{-1} \|_1 = \max \left\{ \frac{1}{\beta_k}\left(1 - 2\frac{\alpha_k}{\beta_k}\right) + \frac{1}{\beta_k} \frac{\sum_{l=1}^{N} \frac{\alpha_l}{\beta_l}}{1+\sum_{l=1}^{N} \frac{\alpha_l}{\beta_l}} \;\middle|\; k = 1, \dots N \right\}$$

puisqu'on a $\dfrac{\alpha_k}{\beta_k}\alpha = \dfrac{\frac{\alpha_k}{\beta_k}}{1+\sum \frac{\alpha_k}{\beta_k}} < 1$ dans le cas où $x_n < \sqrt{a_n b_n}$

$$\| J^{-1} \|_\infty = \max \left\{ \frac{1}{\beta_k}\left(1 - 2\frac{\alpha_k}{\beta_k}\right) + \frac{\alpha_k}{\beta_k} \frac{\sum_{l=1}^{N} \frac{1}{\beta_l}}{1+\sum_{l=1}^{N} \frac{\alpha_l}{\beta_l}} \;\middle|\; k = 1, \dots N \right\}$$

c-Trous dans une même ligne ou colonne

$\forall n \in [1,N]$ $\quad K_n = \dfrac{x_n (c - \sum_{k=1}^{N} x_k)}{(a+\sum_{k=1}^{N} x_k)(b_n + x_n)} = F_n(x_1 \dots x_N)$

$k \neq n$ $\quad \dfrac{\partial F_n}{\partial x_k} = -\dfrac{x_n(c-a)}{(b_n+x_n)(a+\sum_{k=1}^{N} x_k)^2} = -\alpha_n$

$$\frac{\partial F_n}{\partial x_n} = -\frac{x_n(c-a)}{(b_n+x_n)(a+\sum_{k=1}^{N} x_k)^2} + \frac{(c+\sum_{k=1}^{N} x_k)\, b_n}{(a+\sum_{k=1}^{N} x_k)(b_n+x_n)^2} = \beta_n - \alpha_n$$

J est de la forme

$$J = \begin{pmatrix} \beta_1 - \alpha_1 & \cdots & -\alpha_1 & \cdots & -\alpha_1 \\ \vdots & & & & \\ -\alpha_n & \cdots & \beta_n - \alpha_n & \cdots & -\alpha_n \\ \vdots & & & & \\ -\alpha_N & \cdots & -\alpha_N & \cdots & \beta_N - \alpha_N \end{pmatrix}$$

$$\frac{\alpha_n}{\beta_n} = \frac{X_n(c-a)(b_n+X_n)}{(a+\sum_{k=1}^{N} x_k)(c+\sum_{k=1}^{N} X_k)\,b_n} \quad ; \quad \sum_{n=1}^{N} \frac{\alpha_n}{\beta_n} = \frac{(c-a)}{(a+\sum_{k=1}^{N} X_k)(c+\sum_{k=1}^{N} X_k)} \sum_{n=1}^{N} (X_n + \frac{X_n^2}{b_n})$$

Si $\sum_{k=1}^{N} \frac{\alpha_k}{\beta_k} \neq 1$, soit $\varkappa = \frac{1}{1-\sum_{k=1}^{N} \frac{\alpha_k}{\beta_k}}$

$$J^{-1} = \begin{pmatrix} \frac{1}{\beta_1} + \frac{\alpha_1}{\beta_1}\frac{\alpha}{\beta_1} & \cdots & \frac{\alpha_1}{\beta_1}\frac{\alpha}{\beta_k} & \cdots & \frac{\alpha_1}{\beta_1}\frac{\alpha}{\beta_N} \\ \vdots & & & & \\ \frac{\alpha_k}{\beta_k}\frac{\alpha}{\beta_1} & \cdots & \frac{1}{\beta_k} + \frac{\alpha_k}{\beta_k}\frac{\alpha}{\beta_k} & \cdots & \frac{\alpha_k}{\beta_k}\frac{1}{\beta_N} \\ \frac{\alpha_N}{\beta_N}\frac{\alpha}{\beta_1} & \cdots & \frac{\alpha_N}{\beta_N}\frac{\alpha}{\beta_k} & \cdots & \frac{1}{\beta_N} + \frac{\alpha_N}{\beta_N}\frac{\alpha}{\beta_N} \end{pmatrix}$$

De même

$$\|J^{-1}\|_\infty \leq \max\left\{ \frac{1}{\beta_k} + \frac{\alpha_k}{\beta_k} \frac{\sum_{\ell=1}^{N} \frac{1}{\beta_\ell}}{|1-\sum_{\ell=1}^{N} \frac{\alpha_\ell}{\beta_\ell}|} \;\middle|\; k=1,\dots N \right\}$$

$$\|J^{-1}\|_1 \leq \max\left\{ \frac{1}{\beta_k} + \frac{1}{\beta_k} \frac{\sum_{\ell=1}^{N} \frac{\alpha_\ell}{\beta_\ell}}{|1-\sum_{\ell=1}^{N} \frac{\alpha_\ell}{\beta_\ell}|} \;\middle|\; k=1,\dots N \right\}$$

d-Cas général

Soient N trous $(i_1, j_1) \dots\dots (i_N, j_N)$.

Notons i(n), j(n) les indices du trou n ,

$$\mathcal{P}_n = \{ p \mid p \in [1,N] \quad j(p) = j(n) \}$$

$$\mathcal{Q}_n = \{ q \mid q \in [1,N] \quad i(q) = i(n) \}$$

$\forall n \in [1,N]$

$$K_n = \frac{X_n (c - \sum_{k=1}^{N} X_k)}{(a_n + \sum_{q \in \mathcal{Q}_n} X_q)(b_n + \sum_{p \in \mathcal{P}_n} X_p)} = F_n(X_1 \dots X_N)$$

D'où

$\forall n,n' \in [1,N]$
$n' \notin \mathcal{P}_n \cup \mathcal{Q}_n$ $\quad \frac{\partial F_n}{\partial X_{n'}} = \frac{X_n}{(a_n + \sum_{q \in \mathcal{Q}_n} X_q)(b_n + \sum_{p \in \mathcal{P}_n} X_p)} = \frac{K_n}{c + \sum_{k=1}^{N} X_k}$

$n' \notin \mathcal{Q}_n$
$n' \in \mathcal{P}_n$ $\quad \frac{\partial F_n}{\partial X_{n'}} = \frac{K_n}{(c + \sum_{k=1}^{N} X_k)} - \frac{K_n}{(b_n + \sum_{p \in \mathcal{P}_n} X_p)}$

$n' \in \mathcal{Q}_n$
$n' \notin \mathcal{P}_n$ $\quad \frac{\partial F_n}{\partial X_{n'}} = \frac{K_n}{(c + \sum_{k=1}^{N} X_k)} - \frac{K_n}{(a_n + \sum_{q \in \mathcal{Q}_n} X_q)}$

$$\frac{\partial F_n}{\partial X_n} = \frac{K_n}{c+\sum_{k=1}^{N} X_k} - \frac{K_n}{a_n+\sum_{q\in Q_n} X_q} - \frac{K_n}{b_n+\sum_{p\in P_n} X_p} + \frac{K_n}{X_n}$$

Pour l'inversion de J, on est amené aux systèmes d'équations

$$\forall i\in[1,N] \begin{cases} \forall n\in[1,N] \\ n\neq i \end{cases} \quad \frac{\sum_{k=1}^{N} Y_{ki}}{c+\sum_{k=1}^{N} X_k} - \frac{\sum_{p\in P_n} Y_{pi}}{b_n+\sum_{p\in P_n} X_p} - \frac{\sum_{q\in Q_n} Y_{qi}}{a_n+\sum_{q\in Q_n} X_q} + \frac{\delta_{ni}}{X_n} = 0$$

$$\frac{\sum_{k=1}^{N} Y_{ki}}{c+\sum_{k=1}^{N} X_k} - \frac{\sum_{p\in P_i} Y_{pi}}{b_i+\sum_{p\in P_i} X_p} - \frac{\sum_{q\in Q_i} Y_{qi}}{a_i+\sum_{q\in Q_i} X_q} + \frac{Y_{ii}}{X_i} = \frac{1}{K_i}$$

J^{-1} est alors définie par

$$J^{-1} = \left[Y_{ki} \right]$$

5- Variation de la matrice R en fonction des trous

$$R_J = \begin{bmatrix} \ddots & & 0 \\ & \sqrt{\frac{k_j}{k}} & \\ 0 & & \ddots \end{bmatrix}$$

a- Un seul trou (i_o, j_o)

$$\Delta R_J = \begin{pmatrix} & & \\ & \sqrt{k_j}\left(\frac{1}{\sqrt{c+X_1}} - \frac{1}{\sqrt{c+X_2}}\right) & \\ & \vdots & \\ \cdots & \cdots & \sqrt{\frac{b+X_1}{c+X_1}} - \sqrt{\frac{b+X_2}{c+X_2}} \end{pmatrix} j_o$$

$$\Delta R_J = \begin{pmatrix} & & \\ & \dfrac{\sqrt{k_j}\,(X_2-X_1)}{\sqrt{(c+X_1)(c+X_2)}\left(\sqrt{c+X_1}+\sqrt{c+X_2}\right)} & \\ & \vdots & \\ \cdots & \cdots & \dfrac{(X_1-X_2)(c-b)}{\sqrt{(c+X_1)(c+X_2)}\left[\sqrt{(c+X_1)(b+X_2)}+\sqrt{(c+X_2)(b+X_1)}\right]} \end{pmatrix}$$

$$\|\Delta R_2\|_2 \leq \frac{|\Delta x|}{2(c+\bar{x})^{3/2}} \max\left\{ \max_{j\neq j_0} k_j \,,\, \frac{c-b}{\sqrt{b+\bar{x}}} \right\} \text{ si}$$

en particulier $$\|\Delta R_2\|_2 \leq \frac{|\Delta x|}{2c\sqrt{c}} \max\left\{ \max_{j\neq j_0} k_j \,,\, \frac{c-b}{\sqrt{b}} \right\}$$

b- N trous dans des colonnes et lignes distinctes

$$J_n = \{ j_n \mid \exists i_m \ (i_n, j_n) \in \mathcal{N} \}$$

$$\Delta R_2 = \begin{vmatrix} \ddots & & \\ & \sqrt{k_j}\left(\frac{1}{\sqrt{c+\Sigma x_{i1}}} - \frac{1}{\sqrt{c+\Sigma x_{i2}}}\right) & \\ & & \ddots \\ & \sqrt{\frac{b_n+x_{n1}}{c+\Sigma x_{i1}}} - \sqrt{\frac{b_n+x_{n2}}{c+\Sigma x_{i2}}} & \end{vmatrix}$$

$$\Delta R_2 = \begin{vmatrix} \ddots & & \\ & \sqrt{k_j}\,\dfrac{\Sigma x_{i2} - \Sigma x_{i1}}{\sqrt{(c+\Sigma x_{i1})(c+\Sigma x_{i2})}\left(\sqrt{c+\Sigma x_{i1}} + \sqrt{c+\Sigma x_{i2}}\right)} & \\ & & \ddots \\ & \dfrac{(c+\sum_i x_{i1})(x_{n1}-x_{n2}) + (b_n+x_{n1})(\sum_i x_{i1} - \Sigma x_{i2})}{\sqrt{(c+\Sigma x_{i1})(c+\Sigma x_{i2})}\left(\sqrt{c+\Sigma x_{i1}}\sqrt{b_n+x_{n2}} + \sqrt{c+\Sigma x_{i2}}\sqrt{b_n+x_{n1}}\right)} & \end{vmatrix}$$

Si $x_{n1}, x_{n2} \in [\bar{x}_n, x_n]$

$$\|\Delta R_2\| \leq \max\left\{ \max_{j\neq j_n} \frac{\sqrt{k_j}}{2(c+\sum_k \bar{x}_k)^{3/2}} \,,\, \max_{n\in[1,N]} \frac{\max(c+\sum_{i\neq i_n} x_i - b_n \,,\, b_n + x_n)}{2(c+\sum_{k\neq n} \bar{x}_k)^{3/2}\sqrt{b_n+\bar{x}_n}} \right\} \sum_{k=1}^{N} |\Delta x_k|$$

c- N trous dans même colonne j_0

$$\Delta R_2 = \begin{vmatrix} \ddots & & \\ & \sqrt{k_j}\left(\frac{1}{\sqrt{c+\Sigma x_{i1}}} - \frac{1}{\sqrt{c+\Sigma x_{i2}}}\right) & \\ & & \ddots \\ & \sqrt{\frac{b+\Sigma x_{i1}}{c+\Sigma x_{i1}}} - \sqrt{\frac{b+\Sigma x_{i2}}{c+\Sigma x_{i2}}} & \end{vmatrix}$$

Si $x_{n1}, x_{n2} \geq \overline{x_n}$

$$\|\Delta R_j\| \leq \frac{\sum_{k=1}^{N} |\Delta x_k|}{2\left(c + \sum_n \overline{x}_n\right)^{3/2}} \max\left\{ \max_{j \neq j_0} \sqrt{k_j} \,,\, \frac{c-b}{\sqrt{b + \sum_n \overline{x}_n}} \right\}$$

d- N trous dans même ligne i_0

$$\Delta R_j : \begin{pmatrix} \sqrt{k_j}\left(\frac{1}{\sqrt{c+\Sigma x_{i1}}} - \frac{1}{\sqrt{c+\Sigma x_{i2}}}\right) & & \\ & \ddots & \\ & & \sqrt{\frac{b_n + x_{n1}}{c + \sum_i x_{i1}}} - \sqrt{\frac{b_n + x_{n2}}{c + \sum_i x_{i2}}} \end{pmatrix}$$

$$\|\Delta R_j\| \leq \max\left\{ \max_{j \notin J_n} \frac{\sqrt{k_j}}{2\left(c + \sum_k \overline{x}_k\right)^{3/2}} \,,\, \max_{n \in [1,N]} \frac{\max\left(c + \sum_{i \neq n} x_i - b_n,\, b_n + x_n\right)}{2\left(c + \sum_k \overline{x}_k\right)^{3/2} \sqrt{b_n + \overline{x}_n}} \right\} \sum_{k=1}^{N} |\Delta x_k|$$

e- Cas général

$$J_n = \{ j_m \mid \exists i_m \ (i_m, j_m) \in \mathcal{C}^+ \}$$

$$I_n = \{ i_m \mid \exists j_m \ (i_m, j_m) \in \mathcal{C}^+ \}$$

$$\mathcal{P}_n = \{ k \mid j(k) = j(n) \}$$

$$\mathcal{Q}_n = \{ k \mid i(k) = i(n) \}$$

$$\|\Delta R_j\| \leq \max\left\{ \max_{j \notin J_n} \sqrt{k_j} \,,\, \max_{j \in J_n} \frac{\max\left(c + \sum_{k \notin \mathcal{P}_n} x_k - b_n,\, b_n + \sum_{k \in \mathcal{P}_n} x_k\right)}{\sqrt{b_n + \sum_{k \in \mathcal{P}_n} \overline{x}_k}} \right\} \frac{\sum_{k=1}^{N} |\Delta x_k|}{2\left(c + \sum_{k=1}^{N} \overline{x}_k\right)^{3}}$$

6- Etude de la convergence

Précisons d'abord la marche de l'algorithme à étudier.
Notons k(i,j) les valeurs d'une matrice k donnée ayant N trous. Nous supposons connues des valeurs initiales $\{k_o(i,j),\ (i,j) \in \mathcal{N}\}$ et nous notons $k_o(i,j)$ la matrice k dont les "trous" auront été remplis par les valeurs $k_o(i,j)$ initiales. Celles-ci peuvent être obtenues au rang 0 au moyen des algorithmes précédemment étudiés avec $K_n=1$. Au rang r, on prendra comme valeurs initiales les valeurs obtenues pour une approximation d'ordre r-1.

Le calcul de ces valeurs a été étudié au chapitre IV .

Ici, nous les supposerons acquises et nous utiliserons dans la suite l'algorithme 1, comme algorithme ITER. Il est bien entendu qu'on aurait pu en prendre tout autre amenant à la résolution du même système d'équations :

$$(X_i = K_i\ F_i\ (X_1, \ldots X_N) \quad i = 1, \ldots N)$$

A partir de k_o, nous calculons une matrice Q_o, dépendant de ces valeurs initiales k_o et dont nous cherchons les valeurs propres et vecteurs propres.

Alors $\forall (i,j) \in \mathcal{N}$ $\quad k_o(i,j) = \dfrac{k_o(i)\, k_o(j)}{k_o} \left(1 + \sum_{\lambda_o \in \Lambda_o} \sqrt{\lambda_o}\ \varphi^i_{\lambda_o}\ \varphi^j_{\lambda_o}\right)$

Utilisant l'algorithme ITER, on obtient des valeurs $k_1(i,j)$ solutions de

$\forall (i,j) \in \mathcal{N}$ $\quad k_1(i,j) = \dfrac{k_1(i)\, k_1(j)}{k_1} \left(1 + \sum_{\ell=1,r} \sqrt{\lambda_\ell}\ \varphi^i_{\lambda_\ell}\ \varphi^j_{\lambda_\ell}\right)$

Nous voyons donc que pour une approximation de rang 1 nous n'utiliserons que la plus grande valeur propre (la valeur triviale non comprise), puis pour une approximation de rang 2 les 2 plus grandes valeurs propres .....

Cette matrice $k_1(i,j)$ (k(i,j) complètè avec les $k_1(i,j)$) permet de reconstruire une matrice Q_1. On peut ainsi recommencer les calculs en remplaçant Q_0 par Q_1..

L'approximation de rang r est alors obtenue lorsque ce processus converge, s'il converge, i.e lorsque

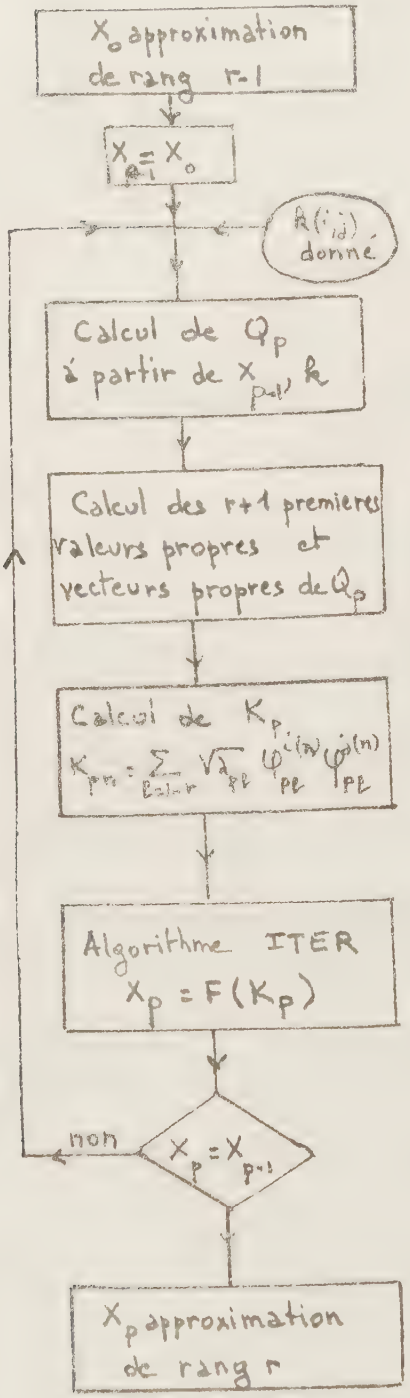
$\forall (i,j) \in \mathcal{N}$ $\quad k_{n+1}(i,j) = k_n(i,j)$

On aura alors

$$\forall (i,j) \in \mathcal{N} \quad \sum_{\ell=1,\ldots r} \sqrt{\lambda_\ell}\ \varphi^i_{\lambda_\ell}\ \varphi^j_{\lambda_\ell} = \sum_{\ell=1,\ldots L} \sqrt{\lambda_\ell}\ \varphi^i_{\lambda_\ell}\ \varphi^j_{\lambda_\ell}$$

avec L = dim Q-1.

Cet algorithme opère selon le schéma suivant, en notant :



X_k la matrice des valeurs des trous

K_n la matrice des coefficients intervenant dans l'algorithme ITER

La convergence de l'algorithme au rang r $(\forall r)$ nous assure de la convergence de l'algorithme "complet", c'est à dire lorsque nous recherchons "la" meilleure approximation et non plus la meille approximation de rang r.

En effet la convergence aura lieu au pire pour l'approximation de rang dimQ-1 = inf(cardI,cardJ)-

De plus,les vecteurs propres φ étant normés et les leurs propres rangées par ordre décroissant,en fai il est évident qu'on pourra s'arrêter lorsque les variations sur K seront inférieures à la précision possible ou désirée.

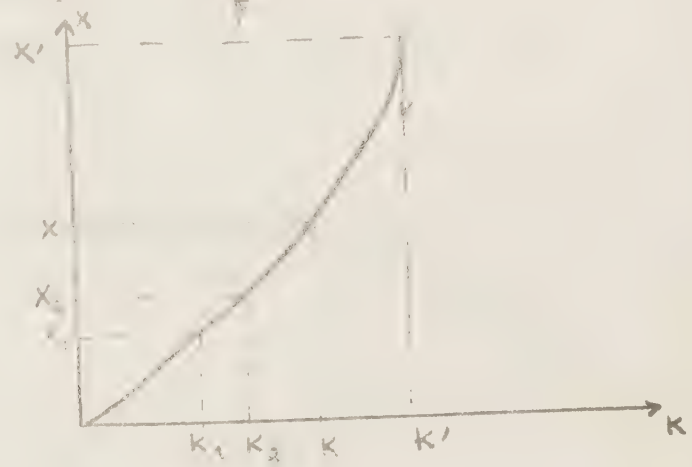
Maintenant, nous nous bornerons à étudier la convergence pour une approximation de rang 1.

Pour une approximation de rang r, le résultat s'en déduit immédiatement.

a-Un seul trou en i_o, j_o

Nous avons deux suites de valeurs X_p, K_p calculées à partir de l'approximation de rang 0, X_o.

X_{p+1} est calculé à partir de K_p selon la courbe



Pour la convergence de l'algorithme ITER, nous devons avoir

$$K_p \leq K' \qquad \text{i.e.} \qquad K_p \leq \frac{\varepsilon^2}{\left(\sqrt{a(c-b)} + \sqrt{b(c-a)}\right)^2}$$

ce qui entraîne $$X_p \leq X' = \frac{ab}{c-(a+b)}\left(1 + \sqrt{\frac{(c-a)(c-b)}{ab}}\right)$$

Considérons deux valeurs successives de X_p que nous noterons X_1 et X_2 pour simplifier l'écriture.

Soit p_j = card J p_i = card I ($p_j \leq$ [illegible]

D'après ce qu'on a vu précédemment

$$\|\Delta G_j\|_2 \leq \frac{\sqrt{p_i}\,|\Delta X|}{(b+X_1)(b+X_2)} \max_{j\in J}\left\{\left|\sum_{i\neq i_o} \frac{k_{ij}k_{i_o}}{k_{i_o}} - \frac{b\,k_{i_o j}}{k_{i_o}}\right|\right\} + \frac{\sqrt{2}\,|\Delta X|\,a}{(a+X_1)(a+X_2)}\left[\sum_{j\in J}\left(\frac{k_{i_o j}}{k_{i_o}}\right)^2\right]^{1/2}$$

Supposons qu'il existe $K, \bar{K}$ tels que $\forall p \geq 1 \qquad \bar{K} \leq K_p \leq K \leq K'$

nous avons alors $\bar{X} \leq X_p \leq X \leq X'$

$$\|\Delta G_j\|_\infty \leq \frac{\sqrt{p_i}\,|\Delta X|}{(b+\bar{X})(b+\bar{X})} \max\left\{\max_{j\neq j_o}\left\{\left|\sum_{i\neq i_o} \frac{k_{ij}k_{i_o}}{k_{i_o}} - b\frac{k_{i_o j}}{a+X_p}\right|\right\}, \left|\sum_{i\neq i_o}\frac{k_{ij}k_{i_o}}{k_{i_o}} - \frac{b\,X_p}{a+X_p}\right|\right\}$$

$$+ \frac{\sqrt{2}\,a\,|\Delta X|}{(a+\bar{X})(a+\bar{X})}\left[\sum_{j\neq j_o}\left(\frac{k_{i_o j}}{k_{i_o}}\right)^2 + \left(\frac{X_p}{a+X_p}\right)^2\right]^{1/2}$$

D'où tenant compte de $\bar{X} \leq X_p \leq X$

$$\|\Delta Q_J\|_2 \leq |\Delta X| \left\{ \frac{\sqrt{P_i}}{(b+\bar{X})^2} \max \left\{ \max_{j \neq j_0} \left(\sum_{i \neq i_0} \frac{k_{ij_0} k_{ij}}{k_i} - \frac{b k_{ij}}{a+X} \; ; \; b \frac{k_{ij}}{a+\bar{X}} - \sum_{i \neq i_0} \frac{k_{ij_0} k_{ij}}{k_i} \right) ; \right.\right.$$

$$\left.\left. \left(\sum_{i \neq i_0} \frac{k_{ij_0}^2}{k_i} - \frac{b\bar{X}}{a+\bar{X}} \right) ; \left(\frac{bX}{a+X} - \sum_{i \neq i_0} \frac{k_{ij_0}^2}{k_i} \right) \right\} + \frac{\sqrt{2}\, a}{(a+\bar{X})^2} \left[\sum_{j \neq j_0} \left(\frac{k_{ij}}{k_j} \right)^2 + \frac{X^2}{(b+X)^2} \right]^{1/2} \right\}$$

Donc pourvu que $X_p \in [\bar{X}, X]$, nous obtenons

$$\|\Delta Q_J\|_2 \leq |\Delta X| \times C'_J(k, X, \bar{X}) \qquad (1)$$

$C'_J(k, X, \bar{X})$ est une constante dépendant du tableau initial k, des valeurs extrêmes $X, \bar{X}$

Opérant de même pour Q_I,

$$\|Q_I\|_2 \leq |\Delta X| \left\{ \frac{\sqrt{P_i}}{(a+\bar{X})^2} \max \left[\max_{i \neq i_0} \left(\sum_{j \neq j_0} \frac{k_{ij} k_{ij}}{k_j} - \frac{a k_{ij_0}}{b+X} \; ; \; \frac{a k_{ij_0}}{b+\bar{X}} - \sum_{j \neq j_0} \frac{k_{ij_0} k_{ij}}{k_j} \right) ; \right.\right.$$

$$\left.\left. \sum_{j \neq j_0} \frac{k_{ij}^2}{k_j} - \frac{a\bar{X}}{b+\bar{X}} \; ; \; \frac{aX}{b+X} - \sum_{j \neq j_0} \frac{k_{ij}^2}{k_j} \right] + \frac{\sqrt{2}\, b}{(b+\bar{X})^2} \left[\sum_{j \neq j_0} \left(\frac{k_{ij_0}}{k_i} \right)^2 + \frac{X^2}{(a+X)^2} \right]^{1/2} \right\}$$

$$\|\Delta Q_I\|_2 \leq |\Delta X| \times C'_I(k, X, \bar{X}) \qquad (2)$$

Q_J est semblable à une matrice symétrique

$$Q_J = R^{-1} S R \qquad \text{avec} \qquad R = \begin{pmatrix} \ddots & & 0 \\ & \sqrt{\frac{k_j}{k}} & \\ 0 & & \ddots \end{pmatrix}$$

$$\|R\|_2 \, \|R^{-1}\|_2 = \sqrt{\frac{\max_{j \in J} (k_j / k)}{\min_{j \in J} (k_j / k)}}$$

$$\|R\|_2 \, \|R^{-1}\|_2 \leq \sqrt{\frac{\max \left(\max_{j \neq j_0} k_j \, , \, b+X \right)}{\min \left(\min_{j \neq j_0} k_j \, , \, b+\bar{X} \right.} \times \frac{c+X}{c+\bar{X}}}$$

Notons $\delta = \inf(1-\lambda_{11}, \lambda_{11}-\lambda_{21})$

si $\|dQ\|_2 \|R\|_2 \|R^{-1}\|_2 \leq \frac{\delta}{2}$

$$|\Delta\lambda| \leq |\Delta x|\, C'_3(k,x,\bar{x}) \sqrt{\frac{\max(\max_{j\neq j_0} k_j, b+x)}{\min(\min_{j\neq j_0} k_j, b+\bar{x})} \times \frac{c+x}{c+\bar{x}}}$$

$$\leq |\Delta x|\, C_3(k,x,\bar{x})$$

avec $$C_3(k,x,\bar{x}) = C'_3(k,x,\bar{x}) \times \sqrt{\frac{\max(\max_{j\neq j_0} k_j, b+x)}{\min(\min_{j\neq j_0} k_j, b+\bar{x})} \times \frac{c+x}{c+\bar{x}}}$$

Soit de même

$$C_1(k,x,\bar{x}) = C'_1(k,x,\bar{x}) \times \sqrt{\frac{\max(\max_{i\neq i_0} k_i, a+x)}{\min(\min_{i\neq i_0} k_i, a+\bar{x})} \times \frac{c+x}{c+\bar{x}}}$$

Nous obtenons

$$|\Delta\lambda| \leq \inf\{C_1(k,x,\bar{x}), C_3(k,x,\bar{x})\}\, |\Delta x|$$

Soit $C = \inf\{C_1(k,x,\bar{x}), C_3(k,x,\bar{x})\}$

Ceci entraîne

si $C|\Delta x| \leq \frac{\delta}{2}$, $|\Delta\lambda| \leq C|\Delta x|$

Etudions maintenant les variations des vecteurs propres :

$$\|dQ\|_2 \|R\|_2 \|R^{-1}\|_2 \leq \delta/8$$

$$\Rightarrow \|\Delta\varphi\|_2 \leq 8\sqrt{7/6} - \sqrt{5/6}\, \frac{\|dQ\|_2}{\delta} \|R\|_2 \|R^{-1}\|_2 \|R_1^{-1}\|_2 (1 + \|R_2\|_2 \|R_1^{-1}\|_2) + \|R_1^{-1}\|_2 \|R_2^{-1}\|_2 \|R_1 - R_2\|_2$$

avec

$$R_3 = \begin{bmatrix} \ddots & & 0 \\ & \sqrt{\frac{k_j}{k_{j_0}}} & \\ 0 & & \ddots \end{bmatrix}$$

$$\|R_3\|_2 \leq \max\left\{\max_{j\neq j_0}\left\{\sqrt{\frac{k_j}{c+\bar{x}}}\right\}, \sqrt{\frac{b-a}{c+\bar{x}}}\right\} = B_3$$

$$\|R_3^{-1}\|_2 \leq \max\left\{\max_{j\neq j_0}\left(\sqrt{\frac{c+x}{k_j}}\right), \sqrt{\frac{c+x}{b-a}}\right\} = B'_3$$

$$\|\Delta R_3\|_2 \leq \frac{|\Delta x|}{2(c+\bar{x})^{3/2}} \max\left\{\max_{j\neq j_0} \sqrt{k_j}, \frac{c-b}{\sqrt{b+\bar{x}}}\right\} = |\Delta x|\, D_3$$

D'où si

$$|\Delta x| C \leq \delta/8$$

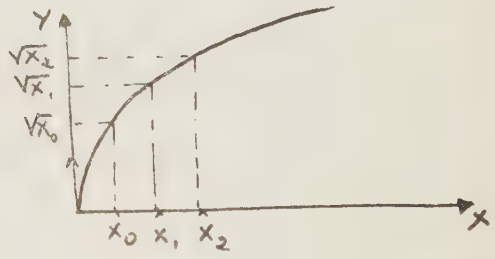
$$\|\Delta\varphi_3\|_2 \leq 8(\sqrt{7/6} - \sqrt{5/6})\, [C_3(k,x,\bar{x}) B'_3 (\ldots) + B_3^2 D_3] \frac{|\Delta x|}{\delta}$$

$$\|\Delta\varphi_3\|_2 \leq A_3 \frac{|\Delta x|}{\delta}$$

De même on définit B_I, B'_I, D_I et

$$\|\Delta\varphi_I\|_2 \leq \frac{8(\sqrt{7/6} - \sqrt{5/6})}{\delta}\left[C_I(R,x,\bar{x})\ (1 + B_I B'_I) + B'_I D_I\right] |\Delta x|\ B'_I \qquad (5)$$

$$\|\Delta\varphi_I\|_2 \leq A_I \frac{|\Delta x|}{\delta}$$



Nous avons $K = 1 + \sqrt{\lambda}\,\varphi_i\,\varphi_j$

Etudions $Y = \sqrt{X}$ $\qquad X \geq X_0$

$$|\sqrt{x_1} - \sqrt{x_2}| \leq \frac{1}{2\sqrt{x_0}} |x_1 - x_2|$$

D'où si

$$\lambda_0 \leq \lambda \leq \lambda_1\ ,\ |\varphi_i| \leq |\bar{\varphi}_i|\ ,\ |\varphi_j| \leq |\bar{\varphi}_j|$$

$$|\Delta K| \leq \frac{1}{2\sqrt{\lambda_0}} |\Delta\lambda|\ |\bar{\varphi}_i|\,|\bar{\varphi}_j| + \sqrt{\lambda_1}\left(|\bar{\varphi}_i|\,|\Delta\varphi_j| + |\bar{\varphi}_j|\,|\Delta\varphi_I|\right) \qquad (6)$$

Or, si $x_1, x_2 \leq x$

$$|x_2 - x_3| \leq |\Delta K| \cdot \frac{(a+x)^2(b+x)^2}{x^2(-c+a+b) + 2abx + abc}$$

$$\leq J(x)\ |\Delta K| \qquad (7)$$

Tenant compte de (1), (2), (3), (4), (5), (6), et (7), nous pouvons obtenir une relation de la forme

$$|x_2 - x_3| \leq \beta\ |x_1 - x_2|$$

β étant un nombre dépendant des intervalles dans lesquels varient

x_n , φ_{in} , φ_{jn} , λ_n , δ_n

Cette condition nous assurera de la convergence de l'algorithme si $\beta < 1$

Notamment, supposons $\beta \leq \frac{1}{2}$ et regardons ce que cela entraîne pour les diverses conditions.

$$|x_2 - x_3| \leq \tfrac{1}{2} |x_1 - x_2|$$

$$\vdots$$

$$|x_n - x_{n+1}| \leq \tfrac{1}{2} |x_{n-1} - x_n|$$

$$\Rightarrow \quad |x_n - x_{n+1}| \leq \frac{1}{2^{n-1}} |x_1 - x_2|$$

D'où $$X_2 - |X_1 - X_2| \frac{1 - \frac{1}{2^{h-1}}}{1 - \frac{1}{2}} \leq X_{h+1} \leq X_2 + |X_1 - X_2| \frac{1 - \frac{1}{2^{h-1}}}{1 - \frac{1}{2}}$$

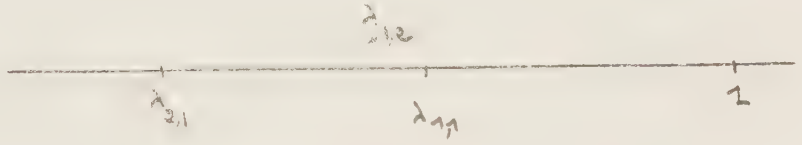
en particulier $$X_2 - |X_1 - X_2| \leq X_{h+1} \leq X_2 + |X_1 - X_2|$$

nous pourrons alors effectuer tous nos calculs en prenant

$$X = X_2 + |X_1 - X_2|$$

$$\bar{X} = X_2 - |X_1 - X_2| \qquad (\bar{X} = 0 \text{ si } X_2 < |X_1 - X_2|)$$

Etudiant de la même façon le spectre de Q_h.



si $\quad C\,|\Delta X| \leq \frac{\delta}{2} \qquad |\Delta\lambda| \leq C\,|\Delta X|$

en particulier si $C\,|\Delta X| \leq \frac{\delta}{8}$, nous obtenons de même

$$\lambda_{1,1} - C|X_1 - X_2|\left(1 + \cdots + \frac{1}{2^{h-2}}\right) \leq \lambda_{1,h} \leq \lambda_{1,1} + C|X_1 - X_2|\left(1 + \cdots + \frac{1}{2^{h-1}}\right)$$

$$\lambda_{1,1} - 2C|X_1 - X_2| \leq \lambda_{1,h} \leq \lambda_{1,1} + 2C|X_1 - X_2|$$

notamment $$\lambda_{1,1} - \frac{\delta}{4} \leq \lambda_{1,h} \leq \lambda_{1,1} + \frac{\delta}{4}$$

De même, d'après les propriétés sur les valeurs propres

$$\lambda_{2,h} \leq \lambda_{2,1} + 2C \cdot |X_1 - X_2| \leq \lambda_{2,1} + \frac{\delta}{4}$$

nous aurons toujours

$$\delta_h = \inf\left(1 - \lambda_{1,h},\ \lambda_{1,h} - \lambda_{2,h}\right) \geq \delta - 4C|X_1 - X_2| \geq \frac{\delta}{2}$$

$h \geq 2$ $\quad |X_{h+1} - X_h|\,C \leq \frac{1}{2^{h-1}}\,C|X_1 - X_2| \leq \frac{\delta}{8} \times \frac{1}{2^{h-1}} \leq \frac{\delta}{8} \times \frac{1}{2^{h-2}} \leq \frac{\delta_h}{8}$

D'après (4)

$$|\varphi_{j,2} - \varphi_{j,1}| \leq A_j \frac{|X_1 - X_2|}{\delta}$$

$$\vdots$$

$$|\varphi_{j,2} - \varphi_{j,h+1}| \leq A_j \frac{|X_h - X_{h+1}|}{\delta_h} \leq \frac{2A_j}{\delta}|X_{h+1} - X_h|$$

$$|\varphi_{j,n+1} - \varphi_{j,2}| \leq \frac{2A_J}{\delta}|x_2 - x_1|$$

Nous obtenons

$$|\varphi_{j,2}| - \frac{2A_J}{\delta}|x_2 - x_1| \leq |\varphi_{j,R}| \leq |\varphi_{j,2}| + \frac{2A_J}{\delta}|x_2 - x_1|$$

De même

$$|\varphi_{i,R}| \leq |\varphi_{i,2}| + \frac{2A_I}{\delta}|x_2 - x_1|$$

Tenant compte de toutes ces inégalités la condition de convergence devient

$$\begin{cases} \bullet \quad J(x)\left\{ \dfrac{c}{2\sqrt{\lambda_{i,1} - \frac{\delta}{4}}}\left(|\varphi_{j,2}| + \frac{2A_J}{\delta}|x_2 - x_1|\right)\left(|\varphi_{i,2}| + \frac{2A_I}{\delta}|x_2 - x_1|\right) \right. \\ \qquad \left. + \sqrt{\lambda_{i,1} + \frac{\delta}{4}}\left[\left(|\varphi_{i,2}| + \frac{2A_I}{\delta}|x_2 - x_1|\right)\frac{A_J}{\delta} + \left(|\varphi_{j,2}| + \frac{2A_J}{\delta}|x_2 - x_1|\right)\frac{A_I}{\delta}\right]\right\} \leq \frac{1}{2} \\ \bullet \quad x = x_2 + |x_1 - x_2| \leq x' \\ \bullet \quad c|x_1 - x_2| \leq \frac{\delta}{8} \end{cases}$$

la 2ème condition entraîne la convergence de l'algorithme ITER

Si nous désirons une approximation de rang r, nous opérerons de la même façon

$$\delta_p = \inf\left(\lambda_{i,p-1} - \lambda_{i,p},\ \lambda_{i,p} - \lambda_{i,p+1}\right) \qquad p = 1, \dots r \text{ avec } \lambda_0 = 1$$

$$\delta = \inf\left(\delta_p \mid p = 1, \dots r\right)$$

D'où les conditions

$$\begin{cases} \bullet \quad J(x)\left\{ \sum_{p=1}^{r}\left[\dfrac{c}{2\sqrt{\lambda_{i,p} - \frac{\delta}{4}}}\left(|\varphi_{j,2}^p| + \frac{2A_J}{\delta_p}|x_2 - x_1|\right)\left(|\varphi_{i,2}^p| + \frac{2A_I}{\delta_p}|x_2 - x_1|\right) \right.\right. \\ \qquad \left.\left. + \sqrt{\lambda_{i,p} + \frac{\delta}{4}}\left(\left(|\varphi_{i,2}^p| + \frac{2A_J}{\delta_p}|x_2 - x_1|\right)\frac{A_I}{\delta_p} + \left(|\varphi_{i,2}^p| + \frac{2A_I}{\delta_p}|x_2 - x_1|\right)\frac{A_J}{\delta_p}\right)\right]\right\} \leq \\ \bullet \quad x \leq x' \\ \bullet \quad c|x - x_2| \leq \frac{\delta}{8} \end{cases}$$

avec

$$X = X_2 + |X_1 - X_2|$$

$$\bar{X} = X_2 - |X_1 - X_2|$$

$$c'_I = \frac{\sqrt{D}}{(a+\bar{X})^2} \max\left\{ \max_{i \neq i_0} \left(\sum_{j \neq j_0} \frac{k_{ij} k_{i_0 j}}{k_j} - \frac{a\, k_{i j_0}}{b+X} , \frac{a\, k_{i_0 j_0}}{b+\bar{X}} - \sum_{j \neq j_0} \frac{k_{i_0 j} k_{ij}}{k_j} \right) , \right.$$

$$\left. \sum_{j \neq j_0} \frac{k_{i_0 j}^2}{k_j} - [illegible] , [illegible] \sum_{j \neq j_0} \frac{k_{ij}^2}{k_j} \right\} + \frac{\sqrt{2}\, b}{(b+\bar{X})^2} \left[\sum_{i \neq i_0} \left(\frac{k_{i j_0}}{k_i} \right)^2 + \left(\frac{X}{a+X} \right)^2 \right]^{1/2}$$

$$C_I = c'_I \sqrt{\frac{\max\left(\max_{i \neq i_0} k_i , a+X \right)}{\min\left(\min_{i \neq i_0} k_i , a+\bar{X} \right)} \times \frac{c+X}{c+\bar{X}}}$$

$$C = \inf(C_I, C_J)$$

$$B_I = \max\left\{ \max_{i \neq i_0} \sqrt{\frac{k_i}{c+\bar{X}}} , \sqrt{\frac{a+X}{c+X}} \right\}$$

$$B'_I = \max\left\{ \max_{i \neq i_0} \sqrt{\frac{c+X}{k_i}} , \sqrt{\frac{c+\bar{X}}{a+\bar{X}}} \right\}$$

$$D_I = \max\left\{ \max_{i \neq i_0} \sqrt{k_i} , \frac{c-a}{\sqrt{a+\bar{X}}} \right\} \times \frac{1}{2\,(c+\bar{X})^{3/2}}$$

$$A_I = 8\left(\sqrt{7/6} - \sqrt{5/6} \right) \left(C_I (1 + B_I B'_I) + B'_I D_I \right) B'_I$$

$$J(x) = \frac{[illegible]}{X^2(-c+a+b) + 2abX + abc}$$

C_J, B_J, B'_J, D_J, A_J sont déterminés par les mêmes formules en permutant i et

b– N trous dans des lignes et colonnes distinctes

Soient $I_n = \{ i_n \mid (i_n, j_n) \in \mathcal{K} \}$

$J_n = \{ j_n \mid (i_n, j_n) \in \mathcal{K} \}$

Nous [illegible] les calculs comme pour un seul trou

$$\| \Delta Q_{j} \|_2 \leq \sum [illegible] \left\{ \frac{\sqrt{[illegible]}}{[illegible]} [illegible] b_n \frac{k_{i_n j}}{k_{i_n}} \Big| \right\} + \frac{\sqrt{2N}\, a_n}{(a_n + X_{n1})(a_n + X_{n2})} \Big(\sum_{j \in J} \Big(\max_n \frac{k_{i_n j}}{k_j} [illegible]$$

Notons de même

$$X_n = X_{n2} + \| X_{n1} - X_{n2} \|_1$$

$$\bar{X}_n = \sup \{ 0, X_{n2} - \| X_{n1} - X_{n2} \|_1 \}$$

$$\| \Delta Q_{j} \|_2 \leq \sum_{n=1}^{N} |\Delta X_n| \left\{ \frac{\sqrt{[illegible]}}{(b_n + \bar{X}_n)^2} [illegible] \left[\max_{j \notin J_n} \Big(\sum_{i \notin I_n} k_{ij_n} \frac{k_{ij}}{k_i} + \sum_{\substack{i_h \in I_n \\ h \neq n}} \frac{k_{i_h j_n} k_{i_h j}}{a_h + \bar{X}_h} - b_n \frac{k_{i_n j}}{a_n + X_n} , \right. \right.$$

$$\frac{b_n k_{[illegible]}}{a_n + \bar{X}_n} [illegible] \sum_{i \notin I_n} \frac{[illegible]}{k_i} - \sum_{\substack{i_h \in I_n \\ h \neq n}} \frac{[illegible]}{a_h + X_h} \Big) , \max_{\substack{j_h \in J_n \\ h \neq n}} \Big(\sum_{i \notin I_n} \frac{k_{ij_n} k_{ij_h}}{k_i} + \sum_{\substack{i_\ell \in I_n \\ \ell \neq (h,n)}} k_{i_\ell j_n} \frac{k_{i_\ell j_h}}{a_\ell + \bar{X}_\ell} [illegible]$$

$$[illegible] \frac{X_h}{a_h + X_h} - b \frac{k_{i_n j_h}}{a_n + X_n} , \frac{b_n k_{i_n j_h}}{a_n + \bar{X}_n} [illegible] k_{i_h j_n} \frac{\bar{X}_h}{a_h + \bar{X}_h} + \sum_{\substack{i_\ell \in I_n \\ \ell \neq (h,n)}} k_{i_\ell j_n} \frac{k_{i_\ell j_h}}{a_\ell + \bar{X}_\ell} - \sum_{i \notin I_n} k_{ij_n} \frac{k_{ij_h}}{k_i} \Big)$$

$$\sum_{i \notin I_n} \frac{k_{ij_n}^2}{k_i} + \sum_{\substack{i_h \in I_n \\ h \neq n}} \frac{k_{i_h j_n}^2}{a_h + \bar{X}_h} - \frac{b_n \bar{X}_n}{a_n + \bar{X}_n} , \frac{b_n X_n}{a_n + X_n} - \sum_{\substack{i_h \in I_n \\ h \neq n}} \frac{k_{i_h j_n}^2}{a_h + X_h} + \sum_{i \notin I_n} \frac{k_{ij_n}^2}{k_i} \Big]$$

$$+ \frac{\sqrt{2N}\, a_n}{(a_n + \bar{X}_n)^2} \left[\sum_{j \notin J_n} \Big(\max_{i \in I_n} \frac{k_{ij}}{k_j} \Big)^2 + \sum_{j_h \in J_n} \Big(\max \Big(\max_{\ell \neq h} \frac{k_{i_\ell j_h}}{b_h + \bar{X}_h} , \frac{X_h}{b_h + X_h} \Big) \Big)^2 \right]^{1/2} \right\}$$

Soit $\| \Delta Q \|_2 \leq \sum_{n=1}^{N} C'_{j_n} \, |\Delta X_n|$

De même

$$\|\Delta q_I\|_2 \le \sum_{n=1}^{N} C'_{In} |\Delta X_n|$$

$$\|R_j\|_2 \le B_j = \frac{1}{\sqrt{c+\sum_{n=1}^{N} X_n}} \max\left(\max_{j\notin J_n} \sqrt{k_j}, \max_{n\in[1,N]} \sqrt{b_n+X_n}\right)$$

$$\|R_j^{-1}\|_2 \le B'_j = \sqrt{c+\sum_{n=1}^{N} X_n} \times \frac{1}{\min\left(\min_{j\notin J_n} \sqrt{k_j}, \min_{n\in[1,N]} \sqrt{a_n+X_n}\right)}$$

$$\frac{\|\Delta R_j\|_2}{\sum_{n=1}^{N} |\Delta X_n|} \le D_j \quad \max\left\{\max_{j\notin J_n} \sqrt{k_j}, \max_{n\in[1,N]} \frac{\max(c+\sum_{i\neq n} X_i - b_n,\ b_n+X_n)}{\sqrt{b_n+X_n}}\right\} \frac{1}{2\left(c+\sum_{n=1}^{N} X_n\right)^{3/2}}$$

nous obtenons de même B_I, B'_I, D_I en changeant j en i.

Soient,

$$C'_j = \max_{n\in[1,N]} C'_{jn} \; ; \quad C'_I = \max_{n\in[1,N]} C'_{In} \; ; \; C_j = C'_j B_j B'_j \; ; \; C_I = C'_I B_I B'_I$$

$$B_j B'_j \|\Delta q_j\|_2 \le C_j \sum_{n=1}^{N} |\Delta X_n|$$

$$B_I B'_I \|\Delta q_I\|_2 \le C_I \sum_{n=1}^{N} |\Delta X_n| \qquad C = \inf(C_j, C_I)$$

D'où si

$$C \sum_{n=1}^{N} |\Delta X_n| \le \frac{\varepsilon}{8}$$

$$\|\Delta\varphi_j\|_2 \le 8\left(\sqrt{7/6} - \sqrt{5/6}\right)\left(C_j(1+B_jB'_j) + B'_j D_j\right) B'_j \frac{\sum_{n=1}^{N} |\Delta X_n|}{\delta}$$

$$\le A_j \frac{\sum_{n=1}^{N} |\Delta X_n|}{\delta}$$

de même

$$\|\Delta q_I\|_2 \le A_I \frac{\sum_{n=1}^{N} |\Delta X_n|}{\delta}$$

avec

$$A_I = 8\left(\sqrt{7/6} - \sqrt{5/6}\right)\left(C_I(1+B_IB'_I) + B'_I D_I\right) B'_I$$

Les vecteurs colonnes (X_h), (K_h) sont liés par

$$\|\Delta X_h\|_1 \le \|J^{-1}\|_1 \, \|\Delta K_h\|_1$$

Or

$$\|J^{-1}\|_1 = \max\left\{\frac{1}{\beta_h}\left(1 - 2\frac{\alpha_h}{\beta_h}\right) + \frac{1}{\beta_h} \frac{\sum_{k=1}^{N} \frac{\alpha_k}{\beta_k}}{1+\sum_{k=1}^{N} \frac{\alpha_k}{\beta_k}} \;\middle|\; h = 1, \dots N\right\}$$

avec

$$\alpha_h = \frac{X_h}{(a_h+X_h)(b_h+X_h)}$$

$$\beta_h = \left(c - \sum_{k=1}^{N} X_k\right) \frac{a_h b_h - X_h^2}{(a_h+X_h)^2 (b_h+X_h)^2}$$

$$\sum_{h=1}^{N} \frac{\alpha_h}{\beta_h} = \sum_{h=1}^{N} \frac{X_h (a_h + X_h)(b_h + X_h)}{(c + \sum_{i=1}^{N} X_i)(a_h b_h - X_h^2)}$$

en tenant compte du fait que

$\bar{X}_n \leq X_{nk} \leq X_n$, nous obtenons en supposant $\boxed{\bar{X}_n \leq \sqrt{a_n\, b_n}}$

$$\|J^{-1}\|_1 \leq \max_{n \in [1,N]} \left\{ \frac{(a_n + X_n)^2 (b_n + X_n)^2}{(c + \sum_{h=1}^{N} \bar{X}_h)(a_n b_n - X_n^2)} \left[2 - \frac{2\bar{X}_n (a_n + \bar{X}_n)(b_n + \bar{X}_n)}{(c + \sum_{h=1}^{N} X_h)(a_n b_n - \bar{X}_n^2)} \right] - \frac{c + \sum_{h=1}^{N} \bar{X}_h}{c + \sum_{h=1}^{N} \bar{X}_h + \sum_{h=1}^{N} X_h \frac{(a_h + X_h)(b_h + X_h)}{a_h b_h - X_h^2}} \right.$$

De plus, nous avons si $c \sum_{h=1}^{N} |\Delta X_h| \leq \frac{\delta}{8}$

$$|\Delta K_n| \leq \left\{ \frac{c}{2\sqrt{\lambda_1 - \delta/4}} \left(|\varphi_{j,n}| + \frac{2A_J}{\delta} \|X_2 - X_1\|_1 \right) \left(|\varphi_{i,n}| + \frac{2A_I}{\delta} \|X_2 - X_1\|_1 \right) \right.$$

$$\left. + \sqrt{\lambda_1 + \delta/4} \left[\left(|\varphi_{i,n}| + \frac{2A_I}{\delta} \|X_2 - X_1\|_1 \right) \frac{A_J}{\delta} + \left(|\varphi_{j,n}| + \frac{2A_J}{\delta} \|X_2 - X_1\|_1 \right) \frac{A_I}{\delta} \right] \right\} \sum_{h=1}^{N} |\Delta X_h|$$

La condition de convergence s'écrit

$$\begin{cases} \bullet\ \|J^{-1}\|_1 \left\{ \frac{c}{2\sqrt{\lambda_1 - \delta/4}} \sum_{h=1}^{N} \left[\left(|\varphi_{j,n}| + \frac{2A_J}{\delta} \|X_2 - X_1\|_1 \right) \left(|\varphi_{i,n}| + \frac{2A_I}{\delta} \|X_2 - X_1\|_1 \right) \right] \right. \\ \qquad \left. + \frac{\sqrt{\lambda_1 + \delta/4}}{\delta} \left[A_J \sum_{h=1}^{N} \left(|\varphi_{i,n}| + \frac{2A_I}{\delta} \|X_2 - X_1\|_1 \right) + A_I \sum_{h=1}^{N} \left(|\varphi_{j,n}| + \frac{2A_J}{\delta} \|X_2 - X_1\|_1 \right) \right] \right\} \leq \\ \bullet\ c \|X_2 - X_1\|_1 \leq \delta/8 \end{cases}$$

Il reste à s'assurer que l'algorithme ITER reste convergent pour

$$K \in \left[\inf (0, K_2 - \|K_2 - K_1\|_1),\ K_2 + \|K_2 - K_1\|_1 \right]$$

pour $N \geq 3$, et ($N=2$ $K_n \geq \frac{1}{2}$) il suffit de vérifier la condition de convergence pour

$K_n = K_{2n} + \|K_2 - K_1\|_1$, ie

$$\forall n \in [1,N] \qquad \left[c - K_n (a_n + b_n) \right]^2 \geq a_n b_n K_n \frac{(2K_n + N - 3)^2}{K_n + N - 2}$$

et si $N = 2$, $K_n \leq \frac{1}{2}$, l'algorithme est convergent.

Si nous cherchons une approximation d'ordre r, nous opérerons de la même façon, en utilisant les mêmes notations que pour un point :

$$\delta_p = \inf\left(\lambda_{1,p-1} - \lambda_{1,p} \,,\, \lambda_{1,p} - \lambda_{1,p+1}\right) \qquad p = 1, \dots r \qquad \lambda_0 = 1$$

- $C \, \|X_1 - X_2\|_1 \leq \dfrac{\delta}{8}$

- $$\|J^{-1}\|_1 \sum_{p=1}^{r} \left\{ \frac{C}{2\sqrt{\lambda_{1,p} - \delta/4}} \sum_{n=1}^{N} \left[\left(|\varphi_{jn}^p| + \frac{2A_J}{\delta_p}\|X_2 - X_1\|_1\right)\left(|\varphi_{in}^p| + \frac{2A_I}{\delta_p}\|X_2 - X_1\|_1\right)\right] \right.$$
$$\left. + \frac{\sqrt{\lambda_p - \delta/4}}{\delta_p}\left[A_J \sum_{n=1}^{N}\left(|\varphi_{in}^p| + \frac{2A_I}{\delta_p}\|X_2 - X_1\|_1\right) + A_I \sum_{n=1}^{N}\left(|\varphi_{jn}^p| + \frac{2A_J}{\delta_p}\|X_2 - X_1\|_1\right)\right]\right\} \leq \frac{1}{2}$$

- $\forall n \in [1,N] \qquad \left[c - K_n(a_n + b_n)\right]^2 \geq a_n b_n K_n \dfrac{\left[2K_n + N - 3\right]^2}{K_n + N - 2}$

- $\forall n \in [1,N] \qquad X_n \leq \sqrt{a_n b_n}$

Cette dernière condition peut se trouver réduite en utilisant un algorithme différent de l'algorithme 1.

c-N trous dans une même colonne j_0

Le cas où les N trous se trouvent dans une même ligne s'obtiendra évidemment en intervertissant les rôles de I et J.

Soit $I_n = \{ i_m \mid (i_m, j_0) \in \mathcal{N} \}$

$$\|\Delta Q_j\|_2 \leq \frac{\sqrt{p_j}\sum_{n=1}^{N}|\Delta X_n|}{(b + \sum_n X_{n1})(b + \sum_n X_{n2})} \left\{ \max_{\substack{j \neq J \\ m \in [1,N]}} \left(\left| \sum_{i \notin I_n} \frac{k_{ij_0} k_{ij}}{k_i} - b\frac{k_{i_m j}}{k_{i_m}} \right| \right) + \max_{j \in J}\left(\max_{n,n'} \left| \frac{k_{i_n j}}{k_{i_n}} - \frac{k_{i_{n'} j}}{k_{i_{n'}}} \right| \right) \sum X_{n1} \right.$$
$$\left. + \sqrt{2N}\left[\sum_{j \in J} \left(\max_{n \in [1,N]} \frac{k_{i_n j}}{k_j} \right)^2 \right]^{1/2} \left[\sum_{n \in [1,N]} \frac{|\Delta X_n| a_n}{(a_n + X_{n1})(a_n + X_{n2})} \right] \right.$$

De la même façon

$$X_n = X_{n2} + \|X_{n1} - X_{n2}\|_1$$

$$\bar{X}_n = \sup\left\{0, \; X_{n2} - \|X_{n1} - X_{n2}\|_1\right\}$$

$$\|\Delta Q_J\|_2 \le \left\{ \frac{\sqrt{P_j}}{\left(b+\sum_{n=1}^{N}\bar{X}_m\right)^2} \max_{m\in[1,N]} \left[\max_{j\neq j_0} \left(\sum_{i\notin I_n} \frac{k_{ij_0}k_{ij}}{k_i} - b\frac{k_{inj}}{a_m+\bar{x}_m}, b\frac{k_{inj}}{a_n+\bar{X}_n} - \sum_{i\notin I_n} \frac{k_{ij_0}k_{ij}}{k_i} \right), \sum_{i\notin I_n} \frac{k_{ij_0}^2}{k_i} - b\frac{\bar{X}_n}{a_n+\bar{X}_n}, \right.\right.$$

$$\left.\left. \frac{bX_n}{a_m+X_m} - \sum_{i\notin I_n} \frac{k_{ij_0}^2}{k_i} \right] + \sqrt{2N} \max_{m\in[1,N]} \left\{ \frac{a_m}{(a_m+\bar{x}_m)^2} \right\} \left[\sum_{j\neq j_0} \left(\max_{m\in[1,N]} \frac{k_{inj}}{k_j} \right)^2 + \left(\max_{m\in[1,N]} \frac{X_m}{b+\sum_{k=1}^{N}\bar{X}_k} \right)^2 \right]^{1/2} \right\} \sum_{k=1}^{N}$$

$$\|\Delta Q_J\|_2 \le C'_J \sum_{k=1}^{N} |\Delta X_k|$$

$$\|\Delta Q_I\|_2 \le \left\{ \sqrt{P_i} \max_{m\in[1,N]} \left[\frac{1}{(a_m+\bar{x}_m)^2} \max_{i\neq I_n} \left(\sum_{j\neq j_0} \frac{k_{inj}k_{ij}}{k_j} - a_m\frac{k_{ij_0}}{b+\sum_{k=1}^{N}\bar{X}_k}, a_m\frac{k_{ij_0}}{b+\sum_{k=1}^{N}\bar{X}_k} - \sum_{j\neq j_0}\frac{k_{inj}k_{ij}}{k_j} \right), \right.\right.$$

$$\left.\left. \frac{1}{(a_m+\bar{x}_m)^2} \max_{k\in[1,N]} \left(\sum_{j\neq j_0} \frac{k_{inj}k_{ij}}{k_j} - a_m\frac{\bar{X}_k}{b+\sum_{k=1}^{N}\bar{X}_k}, a_m\frac{X_k}{b+\sum_{k=1}^{N}\bar{X}_k} - \sum_{j\neq j_0}\frac{k_{inj}k_{ij}}{k_i} \right) \right] \right.$$

$$\left. + \frac{\sqrt{N+1}}{b+\sum_{k=1}^{N}\bar{X}_k} \left[\sum_{i\notin I_n} \left(\frac{k_{ij_0}}{k_i}\right)^2 + \sum_{m=1}^{N} \left(\frac{X_m}{a_m-X_m}\right)^2 \right]^{1/2} \right\} \sum_{k=1}^{N} |\Delta X_k|$$

$$\|\Delta Q_I\|_2 \le C'_I \sum_{k=1}^{N} |\Delta X_k|$$

$$\|R_J\|_2 = B_J = \frac{1}{\sqrt{c+\sum_{n=1}^{N}\bar{X}_n}} \max\left\{ \max_{j\neq j_0}\sqrt{k_j}, \sqrt{b+\sum_{n=1}^{N}\bar{X}_n} \right\}$$

$$\|R_J^{-1}\|_2 = B'_J = \sqrt{c+\sum_{n=1}^{N}X_m} \cdot \frac{1}{\min\left\{ \min_{j\neq j_0}\sqrt{k_j}, \sqrt{b+\sum_{k=1}^{N}\bar{X}_m} \right\}}$$

$$\|R_I\|_2 = B_I = \frac{1}{\sqrt{c+\sum_{n=1}^{N}\bar{X}_n}} \max\left\{ \max_{i\notin I_n}\sqrt{k_i}, \max_{i\in I_n}\sqrt{a_m+X_m} \right\}$$

$$\|R_I^{-1}\|_2 = B'_I = \sqrt{c+\sum_{m=1}^{N}X_m} \cdot \frac{1}{\min\left\{ \min_{i\notin I_n}\sqrt{k_i}, \min_{i\in I_n}\sqrt{a_m+\bar{X}_m} \right\}}$$

$$\frac{\|\Delta R_J\|_2}{\sum_{k=1}^{N}|\Delta X_k|} = D_J = \frac{1}{2\left(c+\sum_{n=1}^{N}\bar{X}_m\right)^{3/2}} \max\left\{ \max_{j\neq j_0}\sqrt{k_j}, \frac{c-b}{\sqrt{b+\sum_{n=1}^{N}\bar{X}_n}} \right\}$$

$$\frac{\|\Delta R_I\|_2}{\sum_{k=1}^{N}|\Delta X_k|} = D_I = \frac{1}{2\left(c+\sum_{n=1}^{N}\bar{X}_m\right)^{3/2}} \max\left\{ \max_{i\notin I_n}\sqrt{k_i}, \max_{m\in[1,N]} \frac{\max\left(c+\sum_{h\neq n}X_h - a_n, a_m+X\right.}{\sqrt{b+\bar{X}_m}} \right.$$

$$C_I = C'_I B_I B'_I \qquad C_J = C'_J B_J B'_J$$

Soit $C = \inf (C_I, C_J)$, si $C \|X_1 - X_2\|_1 \leq \frac{\delta}{8}$

$$\|\Delta \varphi_J\|_2 \leq 8 \left(\sqrt{7/6} - \sqrt{5/6} \right) \left(C_J (1 + B_J B'_J) + B'_J D_J \right) B'_J \frac{\sum_{n=1}^{N} |\Delta X_n|}{\delta}$$

$$\leq A_J \frac{\sum_{n=1}^{N} |\Delta X_n|}{\delta}$$

de même

$$\|\Delta \varphi_I\|_2 \leq A_I \frac{\sum_{n=1}^{N} |\Delta X_n|}{\delta}$$

avec $A_I = 8 \left(\sqrt{7/6} - \sqrt{5/6} \right) \left(C_J (1 + B_I B'_I) + B'_I D_J \right) B'_I$

Dans ce cas

$$\|J^{-1}\|_1 = \max_{h \in [1,N]} \left\{ \frac{1}{\beta_h} + \frac{1}{\beta_h} \frac{\sum_{\ell=1}^{N} \frac{\alpha_\ell}{\beta_\ell}}{1 - \sum_\ell \frac{\alpha_\ell}{\beta_\ell}} \right\}$$ avec $$\beta_h = \frac{\left(c - \sum_{\ell=1}^{N} X_\ell\right) a_h}{\left(b + \sum_{\ell=1}^{N} X_\ell\right) (a_h + X_h)^2}$$

$$\alpha_h = \frac{X_h (c - b)}{\left(b + \sum_{\ell=1}^{N} X_\ell\right)^2 (a_h + X_h)}$$

$$\sum_{\ell=1}^{N} \frac{\alpha_\ell}{\beta_\ell} = \frac{c - b}{\left(c - \sum_{\ell=1}^{N} X_\ell\right)\left(b + \sum_{\ell=1}^{N} X_\ell\right)} \sum_{h=1}^{N} \left(X_h + \frac{X_h^2}{a_h} \right)$$

on voit que si $\boxed{\begin{array}{c} X_n \leq a_n \\ \sum_{n=1}^{N} X_n \leq b \end{array}}$ $\sum_{h=1}^{N} \frac{\alpha_h}{\beta_h} < 1$, notamment $\sum_{h=1}^{N} \frac{\alpha_h}{\beta_h} \neq 1$

$$\|J^{-1}\|_1 = \left(b + \sum_{\ell=1}^{N} X_\ell\right)^2 \max_{m \in [1,N]} \left\{ \frac{(a_m + X_m)^2}{a_m \left[\left(c - \sum_{h=1}^{N} X_h\right)\left(b + \sum_{h=1}^{N} X_h\right) - (c-b) \sum_{h=1}^{N} \left(X_h + \frac{X_h^2}{a_h} \right) \right]} \right\}$$

D'où en tenant compte des majorations :

$$\|J^{-1}\|_1 \leq \left(b + \sum_{\ell=1}^{N} X_\ell\right) \max_{m \in [1,N]} \left\{ \frac{(a_m + X_m)^2}{a_m \left[\left(c - \sum X_\ell\right)\left(b + \sum X_\ell\right) - (c-b) \sum_{h=1}^{N} \left(X_h + \frac{X_h^2}{a_h} \right) \right]} \right\}$$

De plus nous savons que l'algorithme ITER converge si

$$\forall n \in [1,N] \quad \sqrt{b \left(c K_{nh} - \sum_{\ell=1}^{N} K_{\ell h} a_\ell \right)} + \sqrt{\sum_{\ell=1}^{N} (K_{\ell h} a_h)(c - b)} \leq c$$

comme de même

$$\bar{K_n} = \inf \{ 0, K_{n2} - \|K_1 - K_2\|_1 \} \leq K_{nh} \leq K_n = K_{n2} + \|K_1 - K_2\|_1$$

la convergence de l'algorithme ITER est assurée si,

posant $\alpha = \max_n K_n$, $\sqrt{b \left(c\alpha - \sum_{\ell=1}^{N} \bar{K}_\ell a_\ell \right)} + \sqrt{\sum_{\ell=1}^{N} (K_\ell a_\ell)\,(c-b)} \leq c$

On obtient les conditions de convergence suivantes :

$$\|J^{-1}\| \left\{ \frac{c}{2\sqrt{[illegible]}} \sum_{n=1}^{N} \left[[illegible] + \frac{[illegible]}{\delta} \|X_1 - X_2\| \right] \right.$$

$$\left. + \frac{\sqrt{[illegible]}}{\delta} \left[A_J \sum_{n=1}^{N} [illegible] + \frac{[illegible]\, A_I}{\delta} \|X_1 - X_2\| + A_I \sum_{n=1}^{N} [illegible] \|X_1 - X_2\| \right] \right\} \leq \frac{1}{2}$$

- $c\,\|X_2 - X_1\|_1 \leq \delta/8$

et

- $\sqrt{b(\alpha c - \sum_{h=1}^{N} \bar{K}_h a_h)} + \sqrt{\sum_{h=1}^{N} K_h a_h (c-b)} \leq c$

Pour une approximation de rang r, ces conditions deviennent

$$\delta = \inf_p (\delta_p \mid p=1,\dots r) \quad ; \quad \delta_p = \inf(\lambda_{ip-1} - \lambda_{ip}, \lambda_p - \lambda_{ip+1}) \quad p=1,\dots r \quad \lambda_0 = 1$$

- $$\|J^{-1}\| \sum_{p=1}^{r} \left\{ \frac{c}{2\sqrt{\lambda_p - \delta/4}} \sum_{h=1}^{N} \left[c\,|\varphi_{jh}^{p}| + \frac{2A_J}{\delta_p}\|X_2 - X_1\|_1 \right)\left(|\varphi_{ih}^{p}| + \frac{2A_I}{\delta_p}\|X_2 - X_1\|_1 \right)\right]$$

$$+ \frac{\sqrt{\lambda_p + \delta/4}}{\delta_p} \left[A_J \sum_{h=1}^{N} \left(|\varphi_{ih}^{p}| + \frac{2A_I}{\delta_p}\|X_2 - X_1\|_1 \right) + A_I \sum_{h=1}^{N} \left(|\varphi_{jh}^{p}| + \frac{2A_J}{\delta_p}\|X_2 - X_1\|_1 \right) \right] \Bigg\}$$

- $c\,\|X_2 - X_1\|_1 \leq \delta/8$
- $\sqrt{b(\alpha c - \sum_{h=1}^{N} \bar{K}_h a_h)} + \sqrt{\sum_{h=1}^{N} K_h a_h (c-b)} \leq c$
- $X_h \leq a_h \quad ; \quad \sum_{h=1}^{N} X_h \leq b$

Il est à noter que tous les calculs ont été faits en imposant la condition $\beta \leq \frac{1}{2}$ dans l'expression $\|X_2 - X_3\| \leq \beta \|X_2 - X_1\|$ mais on aurait aussi bien pu effectuer les calculs avec un nombre $\gamma\,(\gamma > 1)$ tel que $\beta \leq \frac{1}{\gamma} < 1$

d- Cas général

Dans le cas général, les calculs se conduisent exactement de la même façon.

Le seul problème consiste à déterminer une majoration de la norme de l'inverse du jacobien du système d'équations $K = F(X)$. Dans les cas particuliers considérés, il a été possible de calculer l'inverse de ce jacobien. Lorsqu'on obtient $\|J^{-1}\| \leq M$, il est alors possible de conduire les calculs de la même façon que dans les cas particuliers.

On obtiendra, en utilisant les mêmes notations, les conditions de convergence de la forme

$$c\,\|X_2 - X_1\|_1 \leq \delta/8$$

$$M \sum_{p=1}^{r} \left\{ \frac{c}{2\sqrt{\lambda_p - \delta/4}} \sum_{h=1}^{N} \left[\left(|\varphi_{jh}^{p}| + \frac{2A_J}{p}\|X_2 - X_1\|_1 \right)\left(|\varphi_{ih}^{p}| + \frac{2A_I}{\delta p}\|X_2 - X_1\|_1 \right)\right]\right.$$

$$\left. + \frac{\sqrt{\lambda_p + \delta/4}}{\delta_p} \left[A_J \sum_{h=1}^{N} \left(|\varphi_{ih}^{p}| + \frac{2A_I}{\delta_p}\|X_2 - X_1\|_1 \right) + A_I \sum_{h=1}^{N} \left(|\varphi_{jh}^{p}| + \frac{2A_J}{\delta}\|X_2 - X_1\|_1 \right) \right] \right\} \leq \frac{1}{2}$$

aux quelles on ajoutera les conditions de convergence de l'algorithme ITER.

e-Formules simplifiées

Il est possible de simplifier ces conditions pour en obtenir des conditions suffisantes plus restrictives mais moins lourdes au point de vue du calcul.

Considérons le cas d'un seul trou (i_o, j_o), nous pouvons remarquer que

$$\forall i \in I \qquad \frac{k_{ij}}{k_i} < 1 \qquad\qquad \forall j \in J \qquad \frac{k_{ij}}{k_j} < 1$$

Il est possible de simplifier la majoration de $\|\Delta Q_j\|_2$:

$$\|\Delta Q_j\|_2 \leq \left[\frac{\sqrt{P_i}}{b^2} 2b + \frac{\sqrt{2}}{a} \sqrt{P_j} \right] |\Delta X|$$

$$\leq \left[\frac{2}{b} + \frac{\sqrt{2}}{a} \right] \sqrt{P_j} \; |\Delta X|$$

De même

$$\|\Delta Q_I\|_2 \leq \left[\frac{2}{a} + \frac{\sqrt{2}}{b} \right] \sqrt{P_i} \; |\Delta X|$$

Pour simplifier les calculs dans la suite on pourra de la même façon minorer $\bar{X}$ par 0 et opérer de la même manière.

- lorsque les trous se trouvent dans des colonnes et des lignes distinctes, on obtiendra de façon similaire :

$$\|\Delta Q_j\|_2 \leq \sum_{n=1}^{N} |\Delta X_n| \left[\frac{2}{b_n} + \frac{\sqrt{2N}}{a_n} \right] \sqrt{P_j}$$

$$\|\Delta Q_I\|_2 \leq \sum_{n=1}^{N} |\Delta X_n| \sqrt{P_i} \left[\frac{2}{a_n} + \frac{\sqrt{2N}}{b_n} \right]$$

- lorsque les trous se trouvent dans une même colonne j_o,

$$\|\Delta Q_j\|_2 \leq \sum_{n=1}^{N} |\Delta X_n| \left\{ \frac{2}{b} + \frac{\sqrt{2N}}{\min_{n \in [1,N]} a_n} \right\} \sqrt{P_j}$$

$$\|\Delta Q_I\|_2 \leq \sum_{n=1}^{N} |\Delta X_n| \left\{ \frac{2}{\min_{n \in [1,N]} a_n} + \frac{\sqrt{N+1}}{b} \right\} \sqrt{P_i}$$

On obtiendrait de la même façon une formule simplifiée dans le cas général.

VI - RESULTATS POUR UNE APPROXIMATION DE RANG 0

On a vu précédemment que dans ce cas les algorithmes 1 et 2 peuvent s'appliquer.

Les programmes correspondants ont été utilisés pour 3 tableaux de données différentes.

- un tableau de données climatologiques (fournies par M. MUTOMBO) de dimension 20 x 14 , noté MU
- un tableau sur les préférences des marques de cigarettes (fournies par M. TALLUR), de dimension 30 x 15, noté CI
- un tableau noté CI de teneurs en p.p.m de sédiments marins, (fourni par M. CAZES) de dimension 118 x 14, noté CA.

Les 2 méthodes ont été testées pour différents nombres de "trous". Ces "trous" sont tirés au hasard dans les tableaux complets initiaux.

Nous avons pour ces différents cas :

	card I	card J	N	%
MU	20	14	33	11,8
CI9	30	15	9	2
CI23	30	15	23	5,1
CI55	30	15	55	12,2
CI105	30	15	105	23,3
CA11	118	14	11	0,7
CA41	118	14	41	2,5
CA91	118	14	91	5,5
CA191	118	14	191	11,6

% étant le pourcentage de "trous" par rapport au nombre total de données, i.e. N/cardI x card J.

MU 20x15 33 trous $\varepsilon = 10^{-2}$

METHODE NB D'ITERATIONS		PREMIERE $\mathcal{E}$	DEUXIEME $\mathcal{T}$
1 (1 , 5)	5.11	5.08700	5.08667
2 (1 , 10)	27.08	27.11024	27.10904
3 (1 , 14)	11.47	10.40985	10.40928
4 (2 , 5)	5.15	5.07642	5.07616
5 (2 , 14)	11.77	10.38824	10.38776
6 (3 , 2)	24.12	26.47279	26.47136
7 (3 , 5)	5.16	5.11493	5.11464
8 (3 , 14)	11.69	10.46704	10.46650
9 (4 , 1)	14.80	15.04403	15.04307
10 (4 , 5)	5.17	5.10403	5.10379
11 (5 , 1)	14.80	15.05477	15.05385
12 (6 , 1)	14.80	15.07466	15.07375
13 (7 , 1)	14.80	14.96460	14.96361
14 (7 , 9)	19.80	19.52972	19.52870
15 (8 , 1)	14.80	15.13783	15.13684
16 (8 , 9)	19.85	19.75583	19.75485
17 (9 , 1)	14.80	15.27762	15.27680
18 (10 , 8)	2.10	2.12609	2.12601
19 (11 , 12)	7.90	7.72411	7.72558
20 (12 , 12)	7.95	7.70973	7.71107
21 (13 , 12)	8.00	7.68206	7.68326
22 (14 , 12)	8.00	7.69489	7.69597
23 (15 , 12)	8.15	7.69370	7.69468
24 (16 , 12)	8.16	7.70433	7.70521
25 (17 , 7)	21.42	21.31866	21.31906
26 (17 , 12)	8.17	7.70305	7.70386
27 (18 , 7)	21.45	21.18231	21.18257
28 (18 , 12)	8.18	7.65388	7.65459
29 (19 , 7)	21.47	21.36108	21.36122
30 (19 , 12)	8.21	7.71858	7.71922
31 (20 , 7)	21.51	21.34782	21.34808
32 (20 , 10)	28.06	27.70123	27.70120
33 (20 , 12)	8.28	7.71401	7.71458

CI 9 30x15 9 trous $\varepsilon = 10^{-2}$

	METHODE	PREMIERE	DEUXIEME
NB D'ITERATIONS :		5	4
1	(11 , 1)	40.000	10.155	10.155
2	(26 , 1)	1.000	11.255	11.255
3	(22 , 3)	33.000	29.314	29.314
4	(/ , 5)	13.000	11.549	11.549
5	(21 , 5)	5.000	11.788	11.788
6	(17 , 6)	43.000	34.604	34.605
7	(14 , 8)	3.000	19.496	19.496
8	(20 , 8)	19.000	18.231	18.231
9	(20 , 12)	21.000	11.239	11.239

CI 23 30x15 23 trous $\varepsilon = 10^{-2}$

METHODE			PREMIERE	DEUXIEME
NB D'ITERATIONS :			6	4
1	(11 , 1)	40.000	10.459	10.458
2	(16 , 1)	.200	1.156	1.156
3	(24 , 1)	3.000	11.533	11.533
4	(26 , 1)	1.000	11.591	11.591
5	(6 , 2)	39.000	42.363	42.363
6	(15 , 2)	49.000	39.908	39.909
7	(17 , 3)	15.000	31.103	31.103
8	(22 , 3)	33.000	28.693	28.692
9	(8 , 4)	14.000	15.087	15.086
10	(21 , 4)	44.000	14.142	14.142
11	(22 , 4)	28.000	14.371	14.371
12	(7 , 5)	13.000	11.482	11.481
13	(21 , 5)	5.000	10.833	10.833
14	(15 , 6)	34.000	33.084	33.084
15	(17 , 6)	43.000	36.060	36.059
16	(14 , 8)	3.000	21.843	21.844
17	(20 , 8)	19.000	19.346	19.346
18	(2 , 10)	40.000	33.248	33.248
19	(30 , 10)	42.000	33.064	33.063
20	(14 , 11)	7.000	53.091	53.091
21	(20 , 12)	21.000	11.874	11.874
22	(18 , 14)	46.000	45.131	45.131
23	(20 , 14)	26.000	46.561	46.562

CI 55 55 trous $\epsilon = 10^{-2}$

METHODE			PREMIERE	DEUXIEME
NB D'ITERATIONS =			8	6
1	(1 , 1)	15.CCC	11.819	11.819
2	(11 , 1)	4C.CCC	11.C51	11.C5C
3	(16 , 1)	.2CC	1.142	1.142
4	(24 , 1)	3.CCC	11.242	11.241
5	(26 , 1)	1.CCC	12.329	12.329
6	(6 , 2)	39.CCC	41.366	41.364
7	(15 , 2)	49.CCC	39.591	39.589
8	(17 , 3)	15.CCC	3C.941	3C.94C
9	(2C , 3)	45.CCC	24.696	24.696
1C	(22 , 3)	33.CCC	29.196	29.195
11	(8 , 4)	14.CCC	16.288	16.288
12	(11 , 4)	2.CCC	15.846	15.846
13	(12 , 4)	4.CCC	16.698	16.698
14	(21 , 4)	44.CCC	15.C73	15.C72
15	(22 , 4)	28.CCC	16.CCC	15.999
16	(7 , 5)	13.CCC	11.524	11.523
17	(8 , 5)	11.CCC	11.78C	11.779
18	(1C , 5)	1C.CCC	12.275	12.274
19	(21 , 5)	5.CCC	1C.9CC	1C.9CC
2C	(12 , 6)	39.CCC	36.286	36.285
21	(15 , 6)	34.CCC	32.979	32.978
22	(17 , 6)	43.CCC	36.847	36.845
23	(3C , 6)	29.CCC	34.83C	34.828
24	(8 , 7)	32.CCC	36.312	36.31C
25	(14 , 7)	23.CCC	43.698	43.698
26	(19 , 7)	39.CCC	35.327	35.326
27	(24 , 7)	41.CCC	35.938	35.936
28	(27 , 7)	37.CCC	35.523	35.521
29	(4 , 8)	12.CCC	18.685	18.685
3C	(6 , 8)	24.CCC	18.221	18.221
31	(14 , 8)	3.CCC	22.525	22.525
32	(2C , 8)	19.CCC	15.552	15.552
33	(28 , 8)	24.CCC	17.258	17.257
34	(12 , 9)	8.CCC	1C.C69	1C.C68
35	(2 , 1C)	4C.CCC	36.648	36.648
36	(3 , 1C)	23.CCC	38.379	38.379
37	(1C , 1C)	17.CCC	39.222	39.221
38	(12 , 1C)	33.CCC	38.588	38.587
39	(14 , 1C)	45.CCC	45.298	45.297
4C	(22 , 1C)	21.CCC	36.974	36.973
41	(26 , 1C)	1C.CCC	4C.857	4C.855
42	(3C , 1C)	42.CCC	37.C39	37.C38
43	(1 , 11)	22.CCC	49.C62	49.C61
44	(13 , 11)	47.CCC	42.248	42.247
45	(14 , 11)	7.CCC	56.745	56.744
46	(11 , 12)	1.CCC	12.429	12.429
47	(17 , 12)	4.CCC	13.3CC	13.3CC
48	(2C , 12)	21.CCC	1C.616	1C.616
49	(23 , 12)	1.CCC	13.C8C	13.C8C
5C	(13 , 14)	69.CCC	39.763	39.762
51	(18 , 14)	46.CCC	43.279	43.277
52	(2C , 14)	26.CCC	36.875	36.874
53	(2C , 15)	71.CCC	32.766	32.765
54	(23 , 15)	49.CCC	37.618	37.617
55	(28 , 15)	54.CCC	36.359	36.358

CA 41 118x14 41 trous $\varepsilon = 10^{-2}$

	METHODE	PREMIERE	DEUXIEME
NB D'ITERATIONS :		9	4

1	(36 , 2)	380.000	307.366	307.345
2	(54 , 2)	170.000	455.971	455.940
3	(64 , 2)	310.000	193.971	193.958
4	(109 , 2)	460.000	458.110	458.078
5	(116 , 2)	190.000	224.778	224.763
6	(12 , 3)	550.000	456.400	456.368
7	(4 , 4)	56.000	25.598	25.597
8	(31 , 4)	38.000	41.858	41.856
9	(52 , 4)	6.000	42.548	42.546
10	(66 , 4)	51.000	34.083	34.082
11	(83 , 4)	41.000	21.555	21.554
12	(20 , 5)	20.000	72.838	72.834
13	(35 , 5)	20.000	63.333	63.330
14	(40 , 5)	95.000	87.870	87.866
15	(46 , 5)	70.000	87.483	87.479
16	(50 , 5)	155.000	122.658	122.652
17	(80 , 5)	30.000	111.192	111.186
18	(41 , 6)	9.000	14.284	14.283
19	(83 , 6)	3.000	7.818	7.818
20	(104 , 6)	12.000	13.665	13.664
21	(3 , 7)	24.000	19.731	19.730
22	(25 , 7)	20.000	21.424	21.423
23	(26 , 7)	33.000	23.895	23.894
24	(50 , 7)	42.000	26.488	26.486
25	(71 , 7)	30.000	26.343	26.341
26	(73 , 7)	29.000	21.356	21.354
27	(111 , 7)	22.000	20.910	20.908
28	(44 , 8)	15.000	13.686	13.685
29	(64 , 8)	5.000	5.740	5.740
30	(104 , 8)	10.000	11.818	11.818
31	(107 , 9)	83.000	64.149	64.146
32	(7 , 10)	31.000	48.443	48.440
33	(91 , 10)	54.000	54.929	54.926
34	(41 , 11)	2.000	2.941	2.941
35	(110 , 11)	3.000	2.590	2.589
36	(1 , 12)	1.000	1.256	1.256
37	(56 , 12)	1.000	1.048	1.048
38	(96 , 12)	1.000	1.249	1.248
39	(70 , 13)	70.000	51.896	51.894
40	(61 , 14)	68.000	216.211	216.197
41	(96 , 14)	380.000	435.561	435.531

CA 11 118x14 11 trous $\varepsilon = 10^{-2}$

METHODE			PREMIERE	DEUXIEME
NB D'ITERATIONS :			7	3
1	(54 , | 2)	170.000	457.589	457.594
2	(35 , | 5)	20.000	62.580	62.579
3	(41 , | 6)	9.000	14.217	14.217
4	(104 , | 6)	12.000	13.595	13.594
5	(73 , | 7)	29.000	21.640	21.639
6	(44 , | 8)	15.000	13.662	13.662
7	(64 , | 8)	5.000	6.458	6.458
8	(107 , | 9)	83.000	64.095	64.094
9	(7 , | 10)	31.000	48.411	48.410
10	(1 , | 12)	1.000	1.252	1.252
11	(61 , | 14)	65.000	215.634	215.636

CA 91 118x14 91 trous $\varepsilon = 10^{-2}$

METHODE NB D'ITERATIONS =			PREMIERE 1001	DEUXIEME 1001
1	(65 , | 1)	31C•CCC	153•706	153•704
2	(14 , | 2)	335•CCC	189•825	189•824
3	(16 , | 2)	18C•CCC	533•313	533•311
4	(36 , | 2)	38C•CCC	321•284	321•283
5	(54 , | 2)	17C•CCC	471•849	471•847
6	(55 , | 2)	11C•CCC	215•948	215•947
7	(64 , | 2)	31C•CCC	2CC•642	2CC•64C
8	(79 , | 2)	32C•CCC	464•382	464•38C
9	(81 , | 2)	6CC•CCC	349•272	349•269
1C	(84 , | 2)	28C•CCC	435•341	435•339
11	(92 , | 2)	5CC•CCC	29C•412	29C•411
12	(1C9 , | 2)	46C•CCC	539•773	539•772
13	(116 , | 2)	19C•CCC	232•543	232•542
14	(12 , | 3)	55C•CCC	47C•234	47C•229
15	(29 , | 3)	7C5•CCC	359•165	359•162
16	(1C9 , | 3)	34C•CCC	577•611	577•6C8
17	(118 , | 3)	6C•CCC	239•8C4	239•8CC
18	(4 , | 4)	56•CCC	25•985	25•985
19	(23 , | 4)	45•CCC	39•6C9	39•6C8
2C	(31 , | 4)	38•CCC	42•494	42•494
21	(52 , | 4)	6•CCC	43•195	43•194
22	(66 , | 4)	51•CCC	34•6CC	34•6CC
23	(83 , | 4)	41•CCC	21•969	21•969
24	(9C , | 4)	33•CCC	37•795	37•795
25	(93 , | 4)	43•CCC	35•817	35•816
26	(98 , | 4)	47•CCC	38•475	38•475
27	(2C , | 5)	2C•CCC	75•214	75•213
28	(35 , | 5)	2C•CCC	65•397	65•396
29	(37 , | 5)	45•CCC	96•511	96•51C
3C	(4C , | 5)	95•CCC	9C•739	9C•738
31	(46 , | 5)	7C•CCC	9C•34C	9C•338
32	(5C , | 5)	155•CCC	126•692	126•69C
33	(69 , | 5)	2C•CCC	59•176	59•175
34	(8C , | 5)	3C•CCC	114•826	114•824
35	(81 , | 5)	65•CCC	87•C45	87•C43
36	(41 , | 6)	9•CCC	14•627	14•627
37	(83 , | 6)	3•CCC	8•C4C	8•C4C
38	(1C4 , | 6)	12•CCC	13•995	13•995
39	(113 , | 6)	1C•CCC	13•43C	13•43C
4C	(3 , | 7)	24•CCC	2C•175	2C•175
41	(13 , | 7)	22•CCC	21•719	21•719
42	(16 , | 7)	4C•CCC	28•46C	28•46C
43	(19 , | 7)	28•CCC	2C•C7C	2C•C69
44	(25 , | 7)	2C•CCC	21•89C	21•89C
45	(26 , | 7)	33•CCC	24•41C	24•41C
46	(36 , | 7)	5•CCC	17•145	17•145
47	(5C , | 7)	42•CCC	27•129	27•128
48	(71 , | 7)	3C•CCC	26•731	26•73C
49	(73 , | 7)	29•CCC	21•836	21•836
5C	(83 , | 7)	8•CCC	12•578	12•578
51	(89 , | 7)	5•CCC	8•981	8•981
52	(111 , | 7)	22•CCC	21•38C	21•38C
53	(1C , | 8)	8•CCC	1C•644	1C•644
54	(11 , | 8)	12•CCC	11•476	11•475

55	(37 , 8)	11.CCC	11.362	11.362
56	(44 , 8)	15.CCC	13.939	13.938
57	(45 , 8)	17.CCC	11.373	11.373
58	(64 , 8)	5.CCC	5.887	5.887
59	(1C4 , 8)	1C.CCC	12.C37	12.C37
6C	(2 , 9)	66.CCC	52.438	52.438
61	(6 , 9)	28.CCC	49.296	49.296
62	(6C , 9)	6.CCC	31.257	31.256
63	(1C7 , 9)	83.CCC	65.811	65.811
64	(7 , 1C)	31.CCC	49.22C	49.218
65	(43 , 1C)	67.CCC	6C.541	6C.539
66	(71 , 1C)	88.CCC	7C.68C	7C.677
67	(91 , 1C)	56.CCC	55.811	55.8C9
68	(6 , 11)	2.CCC	2.553	2.553
69	(25 , 11)	4.CCC	2.849	2.849
7C	(41 , 11)	2.CCC	2.978	2.978
71	(42 , 11)	5.CCC	2.478	2.478
72	(53 , 11)	1.CCC	2.64C	2.64C
73	(1CC , 11)	2.CCC	2.581	2.581
74	(11C , 11)	3.CCC	2.622	2.622
75	(115 , 11)	2.CCC	1.556	1.556
76	(118 , 11)	1.CCC	1.556	1.556
77	(1 , 12)	1.CCC	1.263	1.263
78	(17 , 12)	1.CCC	1.122	1.122
79	(26 , 12)	3.CCC	1.327	1.327
8C	(36 , 12)	1.CCC	.932	.932
81	(56 , 12)	1.CCC	1.C54	1.C54
82	(96 , 12)	1.CCC	1.268	1.268
83	(118 , 12)	1.CCC	.65C	.65C
84	(6 , 13)	4C.CCC	5C.873	5C.873
85	(27 , 13)	57.CCC	56.115	56.114
86	(7C , 13)	7C.CCC	53.C55	53.C55
87	(14 , 14)	65.CCC	197.795	197.793
88	(61 , 14)	65.CCC	225.5C5	225.5C3
89	(86 , 14)	36C.CCC	4C7.623	4C7.619
9C	(88 , 14)	17C.CCC	558.C76	558.C72
91	(96 , 14)	38C.CCC	455.161	455.157

CI 105 30x15 105 trous $\varepsilon = 10^{-2}$

METHODE			PREMIERE	DEUXIEME
NB D'ITERATIONS =			10	8
1	(1 , 1)	15.00000	12.39095	12.39[illegible]
2	(6 , 1)	6.00000	11.2[illegible]	11.19973
3	(11 , 1)	4[illegible]	12.12[illegible]7	12.121[illegible]
4	(15 , 1)	13.00000	11.19526	11.19453
5	(16 , 1)	[illegible]	11.1[illegible]	11.12721
6	(21 , 1)	1.00000	11.65659	11.655[illegible]
7	(22 , 1)	[illegible]	12.40550	12.[illegible]
8	(23 , 1)	12.00000	11.20046	11.19941
9	(24 , 1)	5.00000	12.[illegible]	12.68446
10	(26 , 1)	1.00000	13.33947	13.33853
11	([illegible] , 2)	39.[illegible]	38.[illegible]6	38.523[illegible]
12	(15 , 2)	41.00000	38.50569	38.50647
13	(27 , 2)	55.00000	42.10950	42.10561
14	(9 , 3)	45.00000	27.269[illegible]3	27.26707
15	(16 , 3)	2.7[illegible]	2.64111	2.64[illegible]
16	(17 , 3)	18.00000	30.70668	30.70356
17	(20 , 3)	45.[illegible]	23.17795	23.17549
18	(22 , 3)	33.00000	29.58153	29.57916
19	(23 , 3)	35.00000	26.70761	26.70531
20	(27 , 3)	21.00000	29.19267	29.18979
21	([illegible] , 4)	14.[illegible]	14.64625	14.64527
22	(11 , 4)	3.00000	16.10274	16.10164
23	(12 , 4)	4.[illegible]	17.11113	17.11014
24	(16 , 4)	4.40000	1.47123	1.47112
25	(21 , 4)	44.[illegible]	15.48358	15.48263
26	(22 , 4)	28.00000	16.47829	16.47720
27	(3 , 5)	7.00000	12.0636[illegible]	12.06294
28	(7 , 5)	13.00000	11.68040	11.67963
29	([illegible] , 5)	11.00000	10.76728	10.76642
30	(10 , 5)	10.00000	12.47997	12.47927
31	(14 , 5)	22.00000	14.151[illegible]3	14.15100
32	(21 , 5)	5.00000	11.38278	11.38200
33	(24 , 5)	4.[illegible]	12.3[illegible]	12.38640
34	(27 , 5)	14.00000	11.35445	11.35368
35	(11 , 6)	34.[illegible]	34.3[illegible]1	34.32605
36	(12 , 6)	35.00000	36.47[illegible]7	36.4761[illegible]
37	(13 , 6)	41.[illegible]	[illegible]11[illegible]	29.80910
38	(15 , 6)	34.00000	31.70474	31.70117
39	(16 , 6)	3.70000	3.13649	3.13621
40	(17 , 6)	43.00000	36.46593	36.46240
41	(19 , 6)	3[illegible]	[illegible]1	34.39119
42	(20 , 6)	47.[illegible]	27.[illegible]9	27.52[illegible]
43	(30 , 6)	24.00000	34.08932	34.086[illegible]
44	(3 , 7)	46.00000	36.56351	36.56142
45	([illegible] , 7)	32.00000	32.63435	32.63177
46	(14 , 7)	23.00000	42.39250	42.39012
47	(19 , 7)	39.00000	35.94737	35.94[illegible]1
48	(22 , 7)	32.[illegible]	36.71640	36.71365
49	(24 , 7)	41.00000	37.54452	37.541[illegible]5
50	(27 , 7)	37.00000	36.23[illegible]3	36.230[illegible]
51	(4 , 8)	12.00000	18.95995	18.95879
52	(6 , 8)	24.00000	17.46[illegible]	17.46675
53	(11 , 8)	11.00000	18.90625	18.90497
54	(14 , 8)	3.00000	22.60208	22.60071
55	(20 , 8)	19.00000	15.15915	15.15766
56	([illegible] , 8)	24.00000	18.[illegible]	18.30034

57	(1 , 9)	22.00000	10.45523	10.46459
58	(5 , 9)	7.00000	10.07757	10.07711
59	(9 , 9)	3.00000	9.30335	9.30366
60	(12 , 9)	4.00000	10.86945	10.86902
61	(20 , 9)	9.00000	8.20165	8.20089
62	(25 , 9)	1.00000	10.36341	10.36282
63	(2 , 10)	40.00000	38.08896	38.08760
64	(5 , 10)	23.00000	39.14908	39.14763
65	(10 , 10)	17.00000	40.49992	40.49857
66	(12 , 10)	33.00000	40.82249	40.82066
67	(14 , 10)	45.00000	45.92659	45.92439
68	(16 , 10)	2.00000	3.50999	3.50976
69	(18 , 10)	21.00000	39.64702	39.64491
70	(19 , 10)	33.00000	38.49002	38.48773
71	(22 , 10)	21.00000	39.31352	39.31105
72	(24 , 10)	32.00000	40.20036	40.19783
73	(28 , 10)	10.00000	42.27319	42.27084
74	(29 , 10)	23.00000	37.18887	37.18652
75	(29 , 10)	63.00000	37.83420	37.83141
76	(30 , 10)	42.00000	38.14975	38.14719
77	(1 , 11)	22.00000	49.15263	49.14936
78	(6 , 11)	51.00000	44.43144	44.42767
79	(8 , 11)	46.00000	43.74030	43.73674
80	(13 , 11)	47.00000	41.76160	41.75818
81	(14 , 11)	7.00000	57.49123	57.48695
82	(23 , 11)	42.00000	44.43022	44.42645
83	(27 , 11)	33.00000	48.56448	48.55978
84	(29 , 11)	17.00000	47.35951	47.35553
85	(9 , 12)	13.00000	11.88960	11.88820
86	(11 , 12)	1.00000	12.60382	12.60300
87	(15 , 12)	11.00000	11.64011	11.63922
88	(16 , 12)	2.00000	1.15157	1.15148
89	(17 , 12)	4.00000	13.38835	13.38733
90	(20 , 12)	21.00000	10.10575	10.10482
91	(29 , 12)	1.00000	12.41253	12.41159
92	(4 , 13)	14.00000	14.24578	14.24504
93	(10 , 13)	7.00000	15.09523	15.09443
94	(15 , 13)	14.00000	13.11927	13.11853
95	(6 , 14)	55.00000	39.70918	39.70573
96	(13 , 14)	50.00000	37.32291	37.31992
97	(17 , 14)	40.00000	45.65314	45.64948
98	(18 , 14)	46.00000	44.35403	44.35156
99	(20 , 14)	26.00000	34.45934	34.45648
100	(24 , 14)	60.00000	42.32582	42.32242
101	(6 , 15)	35.00000	37.53911	37.53600
102	(8 , 15)	72.00000	34.43628	34.43433
103	(10 , 15)	71.00000	30.35661	30.35420
104	(23 , 15)	43.00000	34.97990	34.97723
105	(26 , 15)	54.00000	36.65047	36.64766

CA 191 118x14 191 trous $\varepsilon = 10^{-2}$

METHODE	PREMIERE	DEUXIEME
NB D'ITERATIONS =	501	501

1	(16 ,	1)1380.0	414.91675	414.93604
2	(18 ,	1)1315.0	288.97534	288.97437
3	(22 ,	1)1300.0	265.96631	265.96582
4	(36 ,	1)1315.0	221.57796	221.57959
5	(38 ,	1)1150.0	328.27051	328.27002
6	(51 ,	1) 40.0	203.41142	203.41364
7	(53 ,	1)1100.0	338.89746	338.89746
8	(65 ,	1)1510.0	182.35612	182.35718
9	(90 ,	1)1150.0	313.36157	313.36377
10	(98 ,	1)1365.0	298.75562	298.75586
11	(101 ,	1)1280.0	293.16162	293.16113
12	(7 ,	2)1630.0	259.26147	259.26294
13	(12 ,	2)1480.0	543.53125	543.56177
14	(14 ,	2)1335.0	178.36537	178.36948
15	(16 ,	2)1180.0	519.63623	519.64233
16	(36 ,	2)1380.0	290.25952	290.26343
17	(39 ,	2)1330.0	157.63979	157.64351
18	(51 ,	2)1909.0	266.46167	266.46631
19	(54 ,	2)1170.0	521.65625	521.66895
20	(55 ,	2)1110.0	278.38989	278.39575
21	(59 ,	2)1320.0	315.47949	316.48193
22	(60 ,	2)1180.0	242.90244	242.90482
23	(64 ,	2)1910.0	136.27628	136.27609
24	(79 ,	2)1320.0	458.61255	458.61670
25	(81 ,	2)1600.0	340.98486	340.98706
26	(84 ,	2)1280.0	425.76147	425.76514
27	(96 ,	2)1500.0	234.02148	234.02368
28	(97 ,	2)1150.0	588.87207	588.87695
29	(109 ,	2)1450.0	528.58740	528.60010
30	(112 ,	2)1300.0	490.68311	490.68701
31	(116 ,	2)1140.0	227.45839	227.45976
32	(12 ,	3)1550.0	596.61646	596.64941
33	(25 ,	3)1705.0	362.89771	362.90015
34	(39 ,	3)1310.0	173.03406	173.03815
35	(44 ,	3)1455.0	530.09326	530.09839
36	(54 ,	3)1350.0	572.59888	572.61279
37	(57 ,	3)1495.0	431.34277	431.34644
38	(61 ,	3)1255.0	225.66228	225.66772
39	(71 ,	3)1451.0	569.47803	569.48315
40	(77 ,	3)1320.0	398.27539	398.27832
41	(109 ,	3)1340.0	580.20463	580.21875
42	(118 ,	3) 60.0	242.39856	242.39964
43	(4 ,	4) 56.0	26.32838	26.32840
44	(16 ,	4) 57.0	49.80254	49.80350
45	(24 ,	4) 45.0	40.03558	40.03613
46	(25 ,	4) 35.0	38.71548	38.71588
47	(31 ,	4) 35.0	42.93307	42.93376
48	(48 ,	4) 20.0	34.80466	34.80516
49	(52 ,	4) 6.0	43.65739	43.65807
50	(63 ,	4) 24.0	26.82675	26.82710
51	(66 ,	4) 51.0	34.97136	34.97180
52	(74 ,	4) 31.0	36.03365	36.03412
53	(83 ,	4) 41.0	24.92848	24.92858
54	(90 ,	4) 42.0	38.18033	38.18074

55	[illegible]	4	[illegible]	33.[illegible]7766	33.[illegible]7811
56	98	4	47.0	37.50751	37.50809
57	102	4	31.0	39.36375	39.36429
58	111	4	34.0	37.77296	37.77344
59	20	5	20.0	71.50706	71.50552
60	35	5	20.0	54.65930	54.65782
61	37	5	45.0	95.42389	95.42180
62	40	5	95.0	89.71301	89.71098
63	46	5	70.0	89.11296	89.11105
64	50	5	155.0	124.23938	124.23656
65	53	5	20.0	111.37444	111.37236
66	69	5	21.0	58.50883	58.50760
67	80	5	36.0	113.52557	113.52313
68	81	5	65.0	85.54597	85.54434
69	93	5	250.0	88.68050	88.67876
70	104	5	220.0	94.36130	94.35937
71	1	6	20.0	11.10005	11.09991
72	17	6	51.0	11.60559	11.60549
73	19	6	70.0	11.24434	11.24422
74	24	6	7.0	11.97125	11.97113
75	34	6	12.0	8.22412	8.22401
76	36	6	3.0	9.17671	9.17664
77	38	6	11.0	13.59510	13.59433
78	41	6	5.0	13.28667	13.28648
79	42	6	10.0	11.09447	11.09437
80	78	6	4.0	12.19298	12.19285
81	83	6	3.0	8.22327	8.22318
82	85	6	3.0	11.54674	11.54660
83	96	6	19.0	12.97772	12.97762
84	104	6	12.0	11.89129	11.89114
85	108	6	45.0	12.98512	12.98497
86	113	6	10.0	12.22162	12.22147
87	3	7	24.0	20.37845	20.37810
88	13	7	22.0	21.79181	21.79135
89	16	7	40.0	28.49873	28.49843
90	19	7	28.0	19.50572	19.50537
91	21	7	13.0	18.40550	18.40518
92	24	7	14.0	20.76672	20.76637
93	25	7	20.0	22.15442	22.15395
94	26	7	33.0	24.65802	24.65756
95	36	7	5.0	15.91902	15.91880
96	41	7	27.0	23.04861	23.04814
97	46	7	23.0	19.48070	19.48032
98	50	7	42.0	27.15959	27.15901
99	57	7	36.0	21.55232	21.55202
100	58	7	13.0	16.99971	16.99942
101	65	7	3.0	13.10092	13.10074
102	71	7	30.0	28.45439	28.45398
103	73	7	25.0	22.05637	22.05597
104	83	7	2.0	14.26511	14.26487
105	86	7	14.0	21.25449	21.25414
106	89	7	5.0	10.56322	10.56301
107	104	7	17.0	20.62810	20.62772
108	111	7	22.0	21.61528	21.61487
109	2	8	21.0	11.37858	11.37824
110	4	8	5.0	8.17715	8.17692
111	7	8	17.0	7.71744	7.71725
112	10	8	8.0	10.60179	10.60150
113	11	8	12.0	11.44130	11.44100
114	13	8	9.0	11.82777	11.82742
115	33	8	20.0	17.00938	17.00893
116	37	8	11.0	11.32212	11.32181
117	44	8	15.0	14.37576	14.37546
118	45	8	17.0	11.33928	11.33899
119	51	8	5.0	7.93180	7.93167
120	64	8	5.0	5.84254	5.84235
121	70	8	9.0	11.22521	11.22491
122	[illegible]	8	8.0	11.48012	11.47985

MU 20x15 33 trous
légère différence pour les trous

1	(1 , 5)	5.11CCC	5.C8852
2	(1 , 10)	27.C8CCC	27.11299
3	(1 , 14)	11.47CCC	10.41211
4	(2 , 5)	5.15CCC	5.C7759
5	(2 , 14)	11.77CCC	10.39CC2
6	(3 , 2)	24.12CCC	26.47935
7	(3 , 5)	5.16CCC	5.11615
8	(3 , 14)	11.69CCC	10.46898
9	(4 , 1)	14.9CCCC	15.C4692
10	(4 , 5)	5.17CCC	5.10481
11	(5 , 1)	14.9CCCC	15.C5767
12	(6 , 1)	14.9CCCC	15.C777C
13	(7 , 1)	14.9CCCC	14.96857
14	(7 , 9)	19.8CCCC	19.53349
15	(8 , 1)	14.9CCCC	15.14182
16	(8 , 9)	19.95CC1	19.75928
17	(9 , 1)	14.8CCCC	15.28C51
18	(10 , 8)	2.1CCCC	2.12668
19	(11 , 12)	7.9CCCC	7.7C193
20	(12 , 12)	7.95CCC	7.68967
21	(13 , 12)	8.CCCCC	7.66394
22	(14 , 12)	8.CCCCC	7.67841
23	(15 , 12)	8.15CCC	7.67875
24	(16 , 12)	8.16CCC	7.69C73
25	(17 , 7)	21.42CCC	21.30536
26	(17 , 12)	8.17CCC	7.69C12
27	(18 , 7)	21.45CCC	21.17117
28	(18 , 12)	8.18CCC	7.64226
29	(19 , 7)	21.47CCC	21.35162
30	(19 , 12)	8.21CCC	7.7C795
31	(20 , 7)	21.5C999	21.33568
32	(20 , 10)	28.C6CCC	27.69168
33	(20 , 12)	8.28CCC	7.7C378

CI 184 30×15 184 trous

Algorithme 1

1	(1 , 2)	65.CCCCC	31.24C84
2	(1 , 4)	7.CCCCC	15.C4C65
3	(1 , 6)	5C.CCCCC	27.17136
4	(1 , 8)	2.CCCCC	15.867C3
5	(1 , 1C)	55.CCCCC	25.6C927
6	(1 , 12)	1.CCCCC	1C.94817
7	(1 , 14)	61.CCCCC	33.83165
8	(2 , 1)	12.CCCCC	13.5C5C2
9	(2 , 3)	25.CCCCC	3C.98721
1C	(2 , 5)	27.CCCCC	11.84629
11	(2 , 7)	27.CCCCC	41.7C532
12	(2 , 9)	7.CCCCC	1C.56CC1
13	(2 , 11)	32.CCCCC	48.67C81
14	(2 , 13)	18.CCCCC	15.74125
15	(2 , 15)	35.CCCCC	43.41C2C
16	(3 , 2)	3C.CCCCC	44.41228
17	(3 , 4)	18.CCCCC	21.38177
18	(3 , 6)	33.CCCCC	38.62668
19	(3 , 8)	14.CCCCC	22.55644
2C	(3 , 1C)	23.CCCCC	36.4C578
21	(3 , 12)	19.CCCCC	15.56371
22	(3 , 14)	37.CCCCC	48.C9435
23	(4 , 1)	14.CCCCC	13.12959
24	(4 , 3)	25.CCCCC	3C.12582
25	(4 , 5)	19.CCCCC	11.517CC
26	(4 , 7)	34.CCCCC	4C.54634
27	(4 , 9)	2C.CCCCC	1C.26657
28	(4 , 11)	29.CCCCC	47.31853
29	(4 , 13)	16.CCCCC	15.3C395
3C	(4 , 15)	32.CCCCC	42.2C436
31	(5 , 2)	64.CCCCC	29.87462
32	(5 , 4)	1C.CCCCC	14.3827C
33	(5 , 6)	41.CCCCC	25.98264
34	(5 , 8)	9.CCCCC	15.17284
35	(5 , 1C)	53.CCCCC	24.48857
36	(5 , 12)	3.CCCCC	1C.469CC
37	(5 , 14)	63.CCCCC	32.35C86
38	(6 , 1)	6.CCCCC	13.56346
39	(6 , 3)	37.CCCCC	31.12143
4C	(6 , 5)	7.CCCCC	11.89763
41	(6 , 7)	35.CCCCC	41.88661
42	(6 , 9)	4.CCCCC	1C.6C595
43	(6 , 11)	51.CCCCC	48.88295
44	(6 , 13)	8.CCCCC	15.8C997
45	(6 , 15)	37.CCCCC	43.89598
46	(7 , 2)	49.CCCCC	34.79312
47	(7 , 4)	3.CCCCC	16.75C52
48	(7 , 6)	52.CCCCC	3C.26C12
49	(7 , 8)	8.CCCCC	17.67C68
5C	(7 , 1C)	62.CCCCC	28.51996
51	(7 , 12)	1.CCCCC	12.19241
52	(7 , 14)	45.CCCCC	37.67638
53	(8 , 1)	12.CCCCC	13.CC629
54	(8 , 3)	28.CCCCC	27.54857
55	(8 , 5)	11.CCCCC	1C.83174
56	(8 , 7)	32.CCCCC	37.C7814
57	(8 , 9)	3.CCCCC	9.38841
58	(8 , 11)	46.CCCCC	43.27153
59	(8 , 13)	3.CCCCC	13.99514
6C	(8 , 15)	72.CCCCC	3F.89525

61	(9 , 2)	37.00000	40.49927
62	(9 , 4)	13.00000	19.49754
63	(9 , 6)	29.00000	35.22255
64	(9 , 8)	19.00000	20.56853
65	(9 , 10)	31.00000	33.19678
66	(9 , 12)	13.00000	14.19173
67	(9 , 14)	52.00000	43.85449
68	(10 , 1)	4.00000	11.25825
69	(10 , 3)	34.00000	25.83223
70	(10 , 5)	10.00000	9.87561
71	(10 , 7)	28.00000	34.76833
72	(10 , 9)	2.00000	8.80356
73	(10 , 11)	59.00000	40.57614
74	(10 , 13)	.00000	13.12341
75	(10 , 15)	82.00000	36.19133
76	(11 , 2)	51.00000	40.70172
77	(11 , 4)	2.00000	19.59486
78	(11 , 6)	36.00000	35.39828
79	(11 , 8)	11.00000	20.67113
80	(11 , 10)	52.00000	33.36218
81	(11 , 12)	1.00000	14.26242
82	(11 , 14)	40.00000	44.07278
83	(12 , 1)	11.00000	14.18013
84	(12 , 3)	25.00000	32.53662
85	(12 , 5)	9.00000	12.43873
86	(12 , 7)	40.00000	43.79239
87	(12 , 9)	8.00000	11.08854
88	(12 , 11)	47.00000	51.10800
89	(12 , 13)	7.00000	16.52977
90	(12 , 15)	25.00000	45.58542
91	(13 , 2)	62.00000	30.67804
92	(13 , 4)	10.00000	14.76912
93	(13 , 6)	41.00000	26.68050
94	(13 , 8)	12.00000	15.58031
95	(13 , 10)	46.00000	25.14569
96	(13 , 12)	3.00000	10.74982
97	(13 , 14)	69.00000	33.21834
98	(14 , 1)	44.00000	11.50417
99	(14 , 3)	5.00000	26.39664
100	(14 , 5)	22.00000	10.09142
101	(14 , 7)	23.00000	35.52856
102	(14 , 9)	46.00000	8.99609
103	(14 , 11)	7.00000	41.46385
104	(14 , 13)	56.00000	13.41061
105	(14 , 15)	12.00000	36.98352
106	(15 , 2)	49.00000	33.23869
107	(15 , 4)	13.00000	16.00179
108	(15 , 6)	34.00000	28.90727
109	(15 , 8)	24.00000	16.88062
110	(15 , 10)	31.00000	27.24425
111	(15 , 12)	11.00000	11.64694
112	(15 , 14)	59.00000	35.99051
113	(16 , 1)	.20000	1.41770
114	(16 , 3)	2.70000	3.25295
115	(16 , 5)	1.20000	1.24360
116	(16 , 7)	3.20000	4.37833
117	(16 , 9)	.30000	1.10862
118	(16 , 11)	4.60000	5.10977
119	(16 , 13)	1.00000	1.65265
120	(16 , 15)	4.00000	4.55765
121	(17 , 2)	63.00000	39.33916
122	(17 , 4)	6.00000	18.93858
123	(17 , 6)	43.00000	34.21252
124	(17 , 8)	13.00000	19.97865

125 (17 , 10) 36.00000 32.24413
126 (17 , 12) 4.00000 13.78436
127 (17 , 14) 40.00000 42.59532
128 (18 , 1) 7.00000 12.06175
129 (18 , 3) 34.00000 27.67610
130 (18 , 5) 5.00000 10.58059
131 (18 , 7) 39.00000 37.25105
132 (18 , 9) 5.00000 9.43225
133 (18 , 11) 62.00000 43.47446
134 (18 , 13) 16.00000 14.06096
135 (18 , 15) 38.00000 38.77722
136 (19 , 2) 57.00000 36.39345
137 (19 , 4) 7.00000 17.52037
138 (19 , 6) 32.00000 31.65047
139 (19 , 8) 26.00000 18.48251
140 (19 , 10) 33.00000 29.82928
141 (19 , 12) 9.00000 12.75198
142 (19 , 14) 51.00000 39.40508
143 (20 , 1) 3.00000 11.06567
144 (20 , 3) 45.00000 25.39062
145 (20 , 5) 11.00000 9.70687
146 (20 , 7) 30.00000 34.17511
147 (20 , 9) 3.00000 8.65342
148 (20 , 11) 56.00000 39.88483
149 (20 , 13) 3.00000 12.90000
150 (20 , 15) 71.00000 35.57562
151 (21 , 2) 10.00000 46.62910
152 (21 , 4) 44.00000 22.44781
153 (21 , 6) 23.00000 40.55182
154 (21 , 8) 35.00000 23.68048
155 (21 , 10) 12.00000 38.21820
156 (21 , 12) 22.00000 16.33820
157 (21 , 14) 17.00000 50.48683
158 (22 , 1) 6.00000 12.18325
159 (22 , 3) 33.00000 27.95505
160 (22 , 5) 16.00000 10.68727
161 (22 , 7) 33.00000 37.62703
162 (22 , 9) .00000 9.52749
163 (22 , 11) 67.00000 43.91371
164 (22 , 13) 6.00000 14.20312
165 (22 , 15) 43.00000 39.16943
166 (23 , 2) 26.00000 43.68393
167 (23 , 4) 27.00000 21.02986
168 (23 , 6) 29.00000 37.99019
169 (23 , 8) 14.00000 22.18460
170 (23 , 10) 21.00000 35.80376
171 (23 , 12) 21.00000 15.30600
172 (23 , 14) 40.00000 47.29709
173 (24 , 1) 3.00000 11.98050
174 (24 , 3) 58.00000 27.49046
175 (24 , 5) 4.00000 10.51017
176 (28 , 3) 42.00000 27.13853
177 (28 , 8) 24.00000 18.94386
178 (29 , 1) 12.00000 12.22210
179 (29 , 5) 10.00000 10.72269
180 (29 , 9) 13.00000 9.56067
181 (29 , 13) 24.00000 14.25479
182 (30 , 8) 17.00000 19.06566
183 (30 , 12) 17.00000 13.15176
184 (30 , 15) 54.00000 38.28868

Comme nous le voyons dans les tableaux précédents, les 2 algorithmes donnent les mêmes résultats.

L'algortithme 2 demande en général moins d'itérations :

	MU	CI9	CI23	CI55	CI105	CA11	CA41	CA91	CA191
g1	8	5	6	8	10	7	9	1001	501
g2	7	4	4	6	8.	3	4	1001	501

(501 et 1001 étant dans chaque cas le nombre maximum d'itérations possibles).

La précision demandée est toujours de 10^{-2} sauf pour CA11 où elle est de 5.10^{-2}.

En raison de cette identité des résultats, les tableaux suivants ne comportent que les résultats obtenus avec l'algorithme 1.

Cet algorithme a aussi été utilisé sur deux tableaux supplémentaires : l'un à partir des données MU mais pour lequel 3 trous ont été remplacés par 3 autres trous, l'autre à partir des données CA avec 184 trous (soit 41 % de trous) non plus tirés au hasard mais pris régulièrement un sur deux en décalant, dans la majeure partie du tableau.

Les tableaux suivants permettent de comparer les résultats obtenus pour le même couple (i,j) manquant dans les différents cas étudiés.

M U

i,j	P(i,j)		
1,5	5,11	5,09	5,12
1,10	27,08	27,11	27,24
1,13	25,28		25,26
1,14	11,47	10,41	
2,5	5,15	5,08	5,11
2,13	25,34		25,22
2,14	11,77	10,39	
3,2	24,12	26,47	26,64
3,5	5,16	5,11	5,15
3,13	25,37		25,43
3,14	11,69	10,47	
4,1	14,80	15,04	15,02
4,5	5,17	5,10	5,10
5,1	14,80	15,05	15,04
6,1	14,80	15,07	15,06
7,1	14,80	14,96	14,94
7,9	19,80	19,53	19,51
8,1	14,8	15,14	15,12
8,9	19,85	19,76	19,74
9,1	14,8	15,28	15,26
10,8	2,1	2,13	2,12
11;12	7,9	7,72	7,71
12,12	7,95	7,71	7,70
13,12	8	7,68	7,67
14,12	8	7,69	7,68
15,12	8,15	7,69	7,68
16,12	8,16	7,70	7,69
17,7	21,42	21,32	21,29
17,12	8,17	7,70	7,69
18,7	21,45	21,18	21,16
18,12	8,18	7,65	7,64
19,7	21,47	21,36	21,33
19,12	8,21	7,72	7,7
20,7	21,51	21,35	21,32
20,10	28,06	27,70	27,68
20,12	8,28	7,71	7,70

		CI 184	CI 105	CI 55	CI 27	CI 9
1,1	15		12,39	11,82		
6,1	6	13,56	11,20			
11,1	40		12,12	11,05	10,46	10,16
15,1	13		11,2			
15,1	0,2	1,42	1,11	1,14	1,16	
21,1	1		11,66			
22,1	6	12,18	12,41			
23,1	12		11,7			
24,1	3	11,98	12,69	11,24	11,53	
26,1	1		13,34	12,23	11,59	11,26
6,2	39		38,33	41,37	42,06	
15,2	49	33,24	38,31	39,59	39,91	
27,2	55		42,11			
9,3	45		27,21			
16,3	2,7	3,25	2,64			
17,3	15		30,71	30,96	31,1	
20,3	45	25,39	27,18	24,7		
22,3	33	27,46	29,58	29,2	28,69	29,31
25,3	35		23,18			
28,3	21		29,19			
8,4	14		14,65	16,29	15,09	
11,4	2	19,59	16,19	15,85		
12,4	4		19,11	16,7		
16,4	4		1,47			
21,4	44	22,45	15,48	15,07	14,14	
22,4	28		16,68	16	16,33	
3,5	7		14,06			
7,5	13		11,68	11,52	11,48	11,55
8,5	11	10,53	10,77	11,78		
10,5	10	9,88	12,48	12,28		
14,5	22	10,09	14,15			
21,5	5		11,38	10,9	10,83	11,79
24,5	4	10,31	12,39			
27,5	14		11,96			
11,6	36	35,4	34,33			
12,6	39		36,48	36,29		
13,6	41	26,68	29,81			
15,6	34	28,91	31,70	32,98	33,08	
16,6	3,7		3,14			
17,6	43	36,71	36,47	36,85	36,06	34,6
19,6	32	31,65	34,39			
10,6	40		27,53			
30,6	29		34,09	34,81		
3,7	46		36,36			
8,7	32	37,08	37,63	36,31		
14,7	23	35,53	42,89	43,7		
19,7	39		35,95	35,33		
22,7	33	37,63	36,72			
24,7	41		37,55	35,94		
27,7	37		36,23	35,52		
4,8	12		18,96	18,69		
6,8	24		17,47	18,22		
11,8	11	20,67	18,91			
14,8	3		22,60	22,53	21,84	19,5
20,8	19		19,16	19,33	19,35	18,23
28,8	24	18,94	18,30	17,26		
1,9	22		10,46			
5,9	7		10,08			
8,9	3	9,39	9,30			

12,9	8	11,09	10,87	10,07		
10,9	3	8,65	8,20			
25,9	1		10,36			
2,10	40		38,09	36,65	33,25	
3,10	23	36,41	39,11	38,38		
10,10	17		40,5	39,22		
12,10	33		40,82	38,59		
14,10	45		45,93	45,3		
16,10	2,2		3,91			
18,10	21		39,69			
19,10	33	29,83	36,49			
22,10	21		39,31	36,97		
24,10	32		40,2			
26,10	10		43,27	40,86		
28,10	23		37,19			
29,10	69		37,83			
30,10	42		38,16	37,04	33,06	
1,11	12		49,15	49,06		
5,11	51	48,88	44,43			
8,11	46	43,27	43,74			
13,11	47		41,76	42,25		
14,11	7	41,46	57,49	56,75	53,09	
25,11	48		44,43			
27,11	33		48,56			
29,11	17		47,36			
9,12	13	14,19	11,69			
11,12	1	14,26	12,60	12,43		
15,12	11	11,65	11,64			
16,12	2,9		1,15			
17,12	4	13,78	13,39	13,3		
20,12	21		10,11	10,62	11,87	11,24
29,12	1		12,41	13,08		
4,13	16	15,30	14,25			
12,13	7	16,53	15,1			
15,13	16		13,12			
6,14	58		39,71			
13,14	69	33,22	37,32	39,76		
17,14	40	42,6	45,65			
18,14	46		44,36	43,28	45,13	
20,14	26		34,46	36,88	46,56	
29,14	60		42,33			
2,15	35	43,41	37,54			
8,15	72	38,6	34,44			
20,15	71	35,48	30,36	32,77		
23,15	49		34,98	37,62		
28,15	54		36,65	36,36		

		CA191	CA91	CA41	CA11
65,1	310	182,36	153,71		
14,2	335	176,37	169,83		
16,2	180	519,01	533,31		
36,2	380	290,26	321,28	307,37	
51,2	170	521,66	471,85	455,97	457,59
55,2	110	278,39	215,95		
64,2	310	196,28	200,64	193,97	
79,2	320	458,61	464,38		
81,2	600	340,99	349,27		
84,2	280	425,76	435,34		
92,2	500	284,02	290,41		
109,2	460	528,59	539,77	458,11	
116,2	190	227,46	232,54	224,78	
12,3	550	596,62	470,23	456,4	
29,3	705	362,9	359,17		
109,3	340	580,21	577,61		
118,3	60	242,4	239,80		
4,4	56	26,33	25,99	25,6	
23,4	45	40,04	36,61		
31,4	38	42,93	42,49	41,86	
52,4	6	43,68	43,2	42,55	
66,4	51	34,97	34,6	34,08	
83,4	41	24,93	21,97	21,56	
90,4	38	38,18	37,8		
93,4	43	35,88	35,82		
98,4	47	37,51	38,48		
20,5	20	71,51	75,21	72,84	
35,5	20	64,37	65,4	63,33	62,58
37,5	45	95,42	96,51		
40,5	95	89,71	90,74	87,87	
46,5	70	89,11	90,34	87,48	
50,5	155	124,24	126,69	122,66	
69,5	20	58,51	59,18		
80,5	30	113,53	114,83	111,19	
81,5	65	85,55	87,05		
41,6	9	13,29	14,63	14,28	14,22
83,6	3	8,22	8,04	7,82	
104,6	12	11,89	14	13,67	13,6
113,6	10	12,22	13,43		
3,7	24	20,38	20,18	19,73	
13,7	22	21,79	21,72		
16,7	40	28,5	28,46		
19,7	20	19,51	20,07		
25,7	20	22,15	21,89	21,42	
26,7	33	24,66	24,41	23,9	
36,7	5	15,92	17,15		
50,7	42	27,16	27,13	26,49	
71,7	30	28,45	26,73	26,34	
73,7	29	22,06	21,84	21,36	21,64
83,7	6	14,27	12,58		
89,7	5	10,56	8,98		
111,7	22	21,62	21,38	20,91	
10,8	8	10,60	10,64		
11,8	12	11,44	11,48		
37,8	11	11,32	11,36		
44,8	15	14,38	13,94	13,69	13,66
45,8	17	11,34	11,37		
64,8	5	5,84	5,89	5,74	6,46

Ces tableaux montrent aussi bien pour CA ou CI que les valeurs calculées pour chaque trou commun sont très proches quel que soit le pourcentage de trous. Or pou CA, les pourcentages varient de 2% à 23,3%, pour CI de 0,7 % à 11,6% . Dans CI184, non tiré au hasard avec 41% de trous, on obtient des résultats très comparables à ceux de CI105.

VII - RESULTATS PAR UNE APPROXIMATION DE RANG r

Dans cette partie, nous ne considérerons pour ITER que l'algorithme 1 puisque nous avons vu que les algorithmes 1 et 2 donnent les mêmes résultats pour une approximation de rang 0.

Cet algorithme a été étudié sur les mêmes tableaux de données soit pour MU, CI9,CI23,CI55,CA11,CA41, CA91 et sur un tableau de mesures des praomys, not PR.

Ce dernier tableau a été étudié pour les cas suivants:

	card I	card J	N	%
PR25	265	20	25	0,5
PR50	265	20	50	0,9
PR100	265	20	100	1,9
PR197	265	20	197	3,7
PR292	265	20	292	5,5

1- Les tableaux MU,CI et CA ont été étudiés en faisant varier tour par tour les valeurs des trous tel qu'il est expliqué dans le chapitre III mais ils ont aussi été traités en utilisant une légère variante de l'algorithme 1, celle pour laquelle le tableau de données se trouve constamment corrigé, après chaqu calcul de la valeur d'un trou. Pour différencier ces deux algorithmes, nous les appelerons l'un par tour, l'autre par point.

En fait, les calculs réalisés montrent que les résultats sont strictement identiques, non seulement finalement mais même à chaque itération.

L'utilisation de ces algorithmes montre que des nombres négatifs peuvent apparaître.

Les nombres k(i,j) devant être tous positifs, on peut soit à la fin remplacer les nombres négatifs trouvés par zéro, soit à chaque itération remplacer les nombres négatifs par zéro.

En fait, ces deux méthodes donnent des résultats si proches que pour la comparaison des analyses effectuées sur les différents tableaux seuls les résultats obtenus en considérant l'algorithme par tour avec remise à zéro à chaque tour ont été utilisés.

En fait, celui-ci demande en général moins de tours d'itération pour chaque rang, par exemple :

r		1	2	3
CI9	à zéro	8	17	11
	négatifs	12	18	12
CI23	à zéro	14	14	
	négatifs	17	>20	
CI55	à zéro	19		
	négatifs	>20		

Pour chaque tableau, on ne cherche que des approximations d'ordre r avec $r \leq 5$.
Dans ces tableaux, la 5ème valeur, toujours très petite en valeur absolue (2.10^{-5}, 7.10^{-3}, 7.10^{-3}), représente un pourcentage d'inertie faible (1%, 2,8%, 3,3%).

Les tableaux suivants donnent les résultats obtenus, avec remise à zéro et avec des nombres négatifs, pour les 6 approximations cherchées. Les valeurs propres obtenues sont indiquées à chaque rang.

MU 33 20x14 33 trous · $\varepsilon = 10^{-4}$
par tour avec remise à zéro

(1 , 5)	5·11	5·086	5·295	5·247	5·255	5·234	5·264
(1 , 10)	27·08	27·104	27·395	26·960	26·966	26·961	27·062
(1 , 14)	11·47	10·407	14·201	17·441	15·666	14·432	13·012
(2 , 5)	5·15	5·075	5·252	5·249	5·238	5·222	5·248
(2 , 14)	13·77	10·386	14·693	15·831	14·245	13·311	12·355
(3 , 2)	24·12	26·446	25·336	25·007	23·656	23·427	23·529
(3 , 5)	5·16	5·114	5·172	5·192	5·216	5·212	5·217
(3 , 14)	13·69	10·464	12·067	12·460	12·489	12·105	11·797
(4 , 1)	14·80	15·040	15·033	14·470	14·876	14·831	15·387
(4 , 5)	5·17	5·103	5·155	5·175	5·203	5·210	5·211
(5 , 1)	14·80	15·051	15·035	14·491	14·775	14·766	15·310
(5 , 1)	14·80	15·071	15·049	14·454	14·592	14·594	15·729
(7 , 1)	14·80	14·960	14·938	14·363	14·275	14·318	11·453
(7 , 9)	19·80	19·525	19·512	19·720	19·657	19·601	19·147
(8 , 1)	14·80	15·133	15·105	14·732	14·521	14·602	14·763
(8 , 9)	19·85	19·751	19·728	19·805	19·740	19·725	19·679
(9 , 1)	14·80	15·274	15·239	15·169	14·910	15·003	15·599
(10 , 8)	2·10	2·126	2·072	2·248	2·215	2·137	2·001
(11 , 12)	7·90	7·725	7·923	7·853	7·898	7·867	7·786
(12 , 12)	7·95	7·710	7·962	7·882	7·997	7·955	7·841
(13 , 12)	8·	7·682	7·976	7·886	7·865	7·832	7·752
(14 , 12)	8·	7·695	8·012	7·913	7·920	7·877	7·791
(15 , 12)	8·15	7·693	8·082	7·956	7·938	7·891	7·797
(16 , 12)	8·16	7·703	8·177	8·019	7·926	7·870	7·760
(17 , 7)	21·41	21·315	21·512	21·470	21·386	21·436	21·439
(17 , 12)	8·17	7·702	8·244	8·062	7·922	7·866	7·767
(18 , 7)	21·45	21·178	21·385	21·252	21·138	20·512	20·331
(18 , 12)	8·18	7·653	8·212	8·012	7·783	7·571	7·049
(19 , 7)	21·47	21·357	21·606	21·487	21·377	21·488	21·506
(19 , 12)	8·21	7·717	8·351	8·128	7·928	7·882	7·827
(20 , 7)	22·51	21·343	21·651	21·435	21·346	21·517	21·533
(20 , 10)	28·06	27·605	27·959	27·883	27·895	27·773	27·707
(20 , 12)	8·28	7·712	8·403	8·160	7·905	7·862	7·835
r		0	1	2	3	4	5

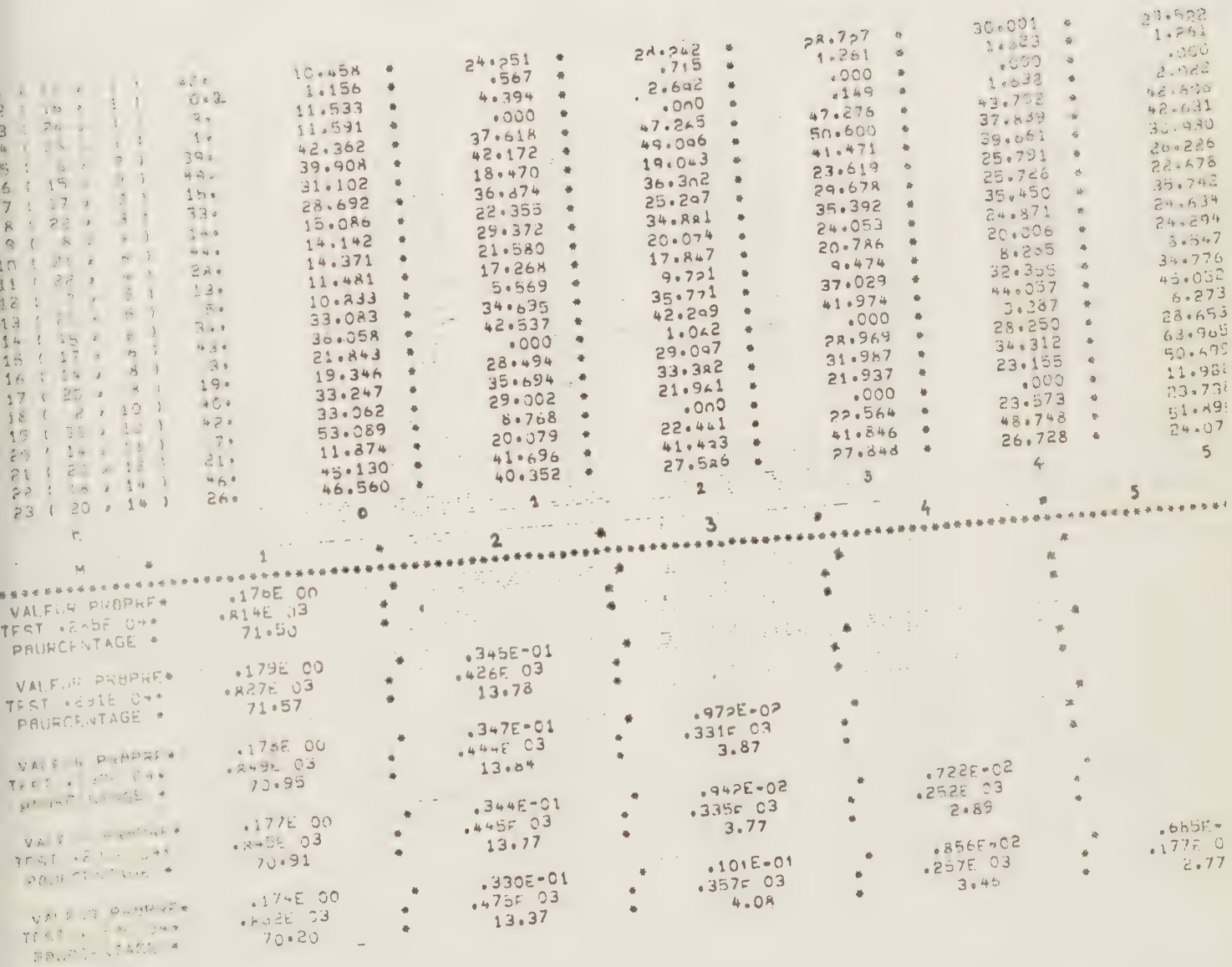
M	1	2	3	4	5
UR PROPRE	·158E-02				
·938E 01	·133E 01				
RCENTAGE	84·06				
UR PROPRE	·243E-02	·241E-03			
·123E 02	·150E 01	·433E 00			
RCENTAGE	87·77	8·70			
UR PROPRE	·189E-02	·225E-03	·589E-04		
·958E 01	·150E 01	·506E 00	·245E 00		
RCENTAGE	24·79	10·09	2·64		
UR PROPRE	·159E-02	·212E-03	·723E-04	·279E-04	
·855E 01	·152E 01	·504E 00	·264E 00	·141E 00	
RCENTAGE	82·22	10·95	3·74	1·44	
UR PROPRE	·133E-02	·271E-03	·138E-03	·744E-04	·380E-04
·448E 01	·238E 01	·115E 01	·548E 00	·219E 00	·508E-01
RCENTAGE	71·46	14·53	7·37	3·99	2·04

CI 9 30x15 9 trous $\varepsilon = 10^{-4}$
par tour avec remise à zéro

11 , 1)	40.	10.155	*	24.095	*	28.245	*	28.156	*	28.458	*	28.626
25 , 1)	1.	11.255	*	.000	*	.000	*	.000	*	.272	*	1.174
22 , 3)	33.	29.314	*	38.027	*	37.988	*	41.436	*	41.294	*	25.395
7 , 5)	13.	11.549	*	17.251	*	17.968	*	18.479	*	27.424	*	27.478
21 , 5)	5.	11.788	*	5.476	*	10.031	*	10.515	*	10.067	*	10.256
17 , 6)	43.	34.604	*	42.175	*	41.686	*	44.391	*	44.259	*	44.051
14 , 8)	3.	19.496	*	.000	*	2.443	*	2.551	*	7.812	*	6.716
20 , 8)	19.	18.231	*	27.815	*	29.143	*	29.087	*	28.339	*	28.072
20 , 12)	21.	11.239	*	19.610	*	22.258	*	22.336	*	23.809	*	23.437
r		0		1		2		3		4		5

	1	2	3	4	5
M *	1 *	2 *	3 *	4 *	5 *
R PROPRE*	.17830728E 00 *				
291E 04*	.88855780E 03 *				
ENTAGE *	71.21 *				
R PROPRE*	.17778677E 00 *	.33353565E-01 *			
291E 04*	.83980620F 03 *	.45137988E 03 *			
ENTAGE *	71.14 *	13.35 *			
R PROPRE*	.17783107E 00 *	.33366664E-01 *	.84205553E-02 *		
291E 04*	.84029571F 03 *	.45147559E 03 *	.35385127E 03 *		
ENTAGE *	71.15 *	13.35 *	3.37 *		
R PROPRE*	.17684969E 00 *	.33549495E-01 *	.85520282E-02 *	.78490872E-02 *	
291E 04*	.84788158E 03 *	.45643499E 03 *	.35665219E 03 *	.26507113E 03 *	
ENTAGE *	70.88 *	13.45 *	3.43 *	3.15 *	
R PROPRE*	.17593047E 00 *	.33607383E-01 *	.85337634E-02 *	.78568240E-02 *	.72813428E-0
290E 04*	.85343232E 03 *	.46179858E 03 *	.36235288E 03 *	.27079569E 03 *	.18594471E 0
ENTAGE *	70.61 *	13.49 *	3.42 *	3.15 *	2.92

CI 23 30x14 23 trous $\varepsilon = 10^{-4}$
par tour avec remise à zéro



30x15 55 trous $\varepsilon = 10^{-4}$

tour avec remise à zéro

1 1	15.	11.818 *	24.619 *	24.800 *	24.936 *	26.021 *	25.809 *	
1 1	40.	11.060 *	24.936 *	28.996 *	28.759 *	34.189 *	32.375 *	
1 1	0.2	1.141 *	.577 *	.713 *	.642 *	1.361 *	1.445 *	
	3.	11.241 *	4.317 *	2.748 *	2.384 *	.000 *	.000 *	
1 1	1.	12.328 *	.000 *	.000 *	.000 *	1.211 *	.130 *	
2 1	38.	41.361 *	37.347 *	47.111 *	45.291 *	45.566 *	45.513 *	
2 2	[illegible]	[illegible] *	41.608 *	48.921 *	52.104 *	43.799 *	53.001 *	
2 [illegible]	[illegible]	[illegible] *	18.895 *	19.892 *	22.134 *	43.115 *	[illegible] *	
3 3	15.	30.937 *	42.301 *	44.579 *	44.758 *	42.662 *	51.120 *	
[illegible]	[illegible]	[illegible] *	[illegible] *	[illegible] *	[illegible] *	25.487 *	[illegible] *	
3 3	38.	29.192 *	37.047 *	36.144 *	34.680 *	21.257 *	22.038 *	
3 4	14.	16.266 *	23.192 *	25.176 *	26.229 *	13.850 *	12.461 *	
4 4	2.	15.845 *	.744 *	5.523 *	4.955 *	9.745 *	10.740 *	
[illegible]	[illegible]	[illegible] *	14.876 *	9.497 *	9.497 *	36.602 *	[illegible] *	
4 4	44.	15.071 *	29.967 *	34.153 *	34.329 *	24.034 *	24.796 *	
4 4	28.	15.938 *	22.077 *	19.655 *	19.764 *	20.929 *	20.315 *	
5 5	13.	11.522 *	17.034 *	17.558 *	20.807 *	56.919 *	48.717 *	
5 5	11.	11.778 *	9.209 *	11.476 *	46.698 *	43.553 *	35.710 *	
5 5	10.	12.273 *	6.912 *	9.829 *	33.072 *	15.666 *	13.768 *	
5 5	5.	10.899 *	5.242 *	9.542 *	12.827 *	38.228 *	40.162 *	
[illegible]	[illegible]	[illegible] *	39.028 *	38.058 *	38.233 *	[illegible] *	[illegible] *	
6 6	34.	32.975 *	34.419 *	35.506 *	34.202 *	42.998 *	34.016 *	
6 6	43.	36.842 *	41.848 *	42.070 *	41.451 *	30.555 *	34.473 *	
6 6	29.	34.825 *	32.896 *	31.443 *	30.892 *	20.804 *	20.145 *	
7 7	32.	36.307 *	37.810 *	36.569 *	18.842 *	24.384 *	27.601 *	
7 7	23.	43.693 *	35.746 *	21.909 *	27.761 *	40.398 *	40.381 *	
[illegible]	[illegible]	[illegible] *	35.637 *	38.843 *	41.019 *	[illegible] *	[illegible] *	
7 7	41.	35.933 *	36.586 *	38.245 *	44.308 *	34.601 *	34.036 *	34.466 *
7 8	37.	35.518 *	34.194 *	33.849 *	34.601 *	9.217 *	12.133 *	
8 8	12.	18.683 *	12.031 *	11.823 *	11.720 *	19.387 *	19.249 *	
8 8	24.	18.219 *	20.736 *	19.606 *	19.777 *	.395 *	1.609 *	
8 8	3.	22.522 *	.000 *	.362 *	2.389 *	2.129 *	.000 *	
8 8	19.	15.551 *	27.040 *	29.295 *	28.681 *	28.256 *	26.923 *	
8 8	24.	17.256 *	26.317 *	25.478 *	25.422 *	25.017 *	23.963 *	
9 9	8.	10.067 *	12.918 *	3.719 *	2.040 *	2.129 *	.000 *	
10 10	40.	36.644 *	38.067 *	35.049 *	35.583 *	35.173 *	38.084 *	
10 10	23.	38.375 *	23.830 *	23.868 *	23.503 *	24.085 *	32.567 *	
10 10	17.	39.217 *	18.276 *	14.473 *	17.224 *	18.697 *	11.373 *	
10 10	33.	38.583 *	41.971 *	46.351 *	46.292 *	46.316 *	48.368 *	
10 10	45.	45.291 *	76.654 *	57.590 *	57.993 *	57.425 *	64.385 *	
10 10	21.	36.969 *	27.379 *	30.020 *	29.624 *	27.779 *	14.121 *	
10 10	10.	40.851 *	11.703 *	7.484 *	6.683 *	4.591 *	.000 *	
10 10	42.	37.033 *	30.885 *	23.470 *	23.194 *	23.156 *	33.096 *	
11 11	22.	49.056 *	22.079 *	22.216 *	20.986 *	21.653 *	21.700 *	
11 11	47.	42.242 *	32.951 *	43.734 *	43.346 *	47.712 *	51.213 *	
11 11	7.	56.737 *	10.544 *	.000 *	.000 *	.000 *	.000 *	
11 12	1.	12.428 *	.514 *	4.684 *	2.635 *	5.332 *	3.803 *	
12 12	4.	13.296 *	5.589 *	8.865 *	7.144 *	6.826 *	.000 *	
12 12	21.	10.615 *	19.023 *	22.672 *	21.079 *	20.240 *	23.578 *	
12 12	1.	13.079 *	5.050 *	2.305 *	1.873 *	1.389 *	5.819 *	
13 14	69.	39.758 *	41.575 *	68.494 *	68.901 *	72.443 *	74.792 *	
13 14	46.	43.273 *	40.543 *	41.426 *	42.854 *	46.922 *	48.876 *	
14 14	26.	36.870 *	38.483 *	27.103 *	29.248 *	31.405 *	27.902 *	
14 15	71.	32.762 *	61.831 *	73.394 *	69.311 *	65.047 *	69.680 *	
15 15	49.	37.613 *	49.751 *	58.969 *	56.651 *	53.194 *	51.461 *	
15 15	54.	36.354 *	60.116 *	57.708 *	54.726 *	55.752 *	55.040 *	
		0	1	2	3	4	5	

M	1	2	3	4	5
R PROPRE	.176E 00				
279E 04	.745E 03				
NTAGE	73.31				
R PROPRE	.179E 00	.346E-01			
290E 04	.804E 03	.400E 03			
NTAGE	72.25	13.94			
R PROPRE	.176E 00	.374E-01	.160E-01		
99E 04	.923E 03	.485E 03	.297E 03		
NTAGE	69.06	14.70	6.28		
		.401E-01	.199E-01	.943E-02	
35E 04	.153E 04	.553E 03	.318E 03	.207E 03	
NTAGE	66.41	15.48	7.69	3.64	
	177E 00	.382E-01	.171E-01	.900E-02	.691E-02
	E 03	.536E 03	.334E 03	.229E 03	.147E 03

CA 11 118x14 11 trous $\varepsilon = 10^{-4}$
par tour avec remise à zéro

1	(54 , 2)	170.	457.591	*	364.315	*	993.488	*	1025.328	*	2579.561	*	1574.974
2	(35 , 5)	20.	62.579	*	25.191	*	24.044	*	33.036	*	.000	*	50.948
3	(41 , 6)	9.	14.217	*	10.423	*	15.045	*	15.122	*	14.521	*	13.820
4	(104 , 6)	12.	13.594	*	16.285	*	14.144	*	14.966	*	15.632	*	14.202
5	(73 , 7)	29.	21.639	*	25.157	*	23.640	*	22.646	*	19.838	*	20.504
6	(44 , 8)	15.	13.662	*	14.490	*	15.088	*	15.080	*	15.777	*	15.286
7	(64 , 8)	5.	6.458	*	6.205	*	5.648	*	5.851	*	5.058	*	6.378
8	(107 , 9)	63.	64.094	*	77.581	*	96.322	*	98.276	*	104.684	*	104.973
9	(7 , 10)	31.	48.410	*	25.079	*	30.707	*	31.280	*	32.563	*	39.701
10	(1 , 12)	1.	1.252	*	1.237	*	1.180	*	1.197	*	1.273	*	1.261
11	(61 , 14)	65.	215.635	*	242.092	*	37.210	*	84.234	*	92.770	*	33.501
	r		0		1		2		3		4		5

M	*	1	*	2	*	3	*	4	*	5
VALEUR PROPRE	*	.126E 00	*		*		*		*	
FST .415E 05	*	.156E 05	*		*		*		*	
POURCENTAGE	*	62.40	*		*		*		*	
VALEUR PROPRE	*	.126E 00	*	.350E-01	*		*		*	
FST .418E 05	*	.156E 05	*	.861E 04	*		*		*	
POURCENTAGE	*	62.11	*	17.27	*		*		*	
VALEUR PROPRE	*	.126E 00	*	.346E-01	*	.162E-01	*		*	
FST .417E 05	*	.156E 05	*	.861E 04	*	.528E 04	*		*	
POURCENTAGE	*	62.25	*	17.11	*	7.99	*		*	
VALEUR PROPRE	*	.137E 00	*	.359E-01	*	.161E-01	*	.122E-01	*	
FST .447E 05	*	.163E 05	*	.884E 04	*	.549E 04	*	.295E 04	*	
POURCENTAGE	*	63.50	*	16.71	*	7.51	*	5.69	*	
VALEUR PROPRE	*	.128E 00	*	.356E-01	*	.162E-01	*	.113E-01	*	.618E-02
FST .426E 05	*	.160E 05	*	.866E 04	*	.531E 04	*	.298E 04	*	.170E 04
POURCENTAGE	*	62.38	*	17.29	*	7.86	*	5.49	*	3.00

CA 41   118x14   41 trous $\varepsilon = 10^{-4}$
par tour avec remise à zéro

(34 , 2)	[illegible]	307.334	617.307	560.677	[illegible]	297.843	[illegible]
(54 , 2)	[illegible]	[illegible]	349.121	989.709	1071.843	[illegible]	[illegible]
(64 , 2)	[illegible]	193.952	234.889	.000	31.233	215.962	[illegible]
(109 , 2)	[illegible]	[illegible]	172.368	839.673	[illegible]	158.077	[illegible]
(116 , 2)	[illegible]	[illegible]	165.054	202.161	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	405.513	494.121	702.972	[illegible]	[illegible]
(4 , 4)	[illegible]	25.596	22.244	29.485	38.930	38.532	[illegible]
(31 , 4)	[illegible]	41.855	47.199	.47.074	43.539	43.977	[illegible]
(22 , 4)	[illegible]	[illegible]	32.545	23.093	[illegible]	42.022	[illegible]
(65 , 4)	[illegible]	[illegible]	30.137	33.297	[illegible]	[illegible]	[illegible]
(83 , 4)	[illegible]	21.553	18.617	20.624	30.248	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	59.320	62.500	65.100	.000	[illegible]
(35 , 5)	[illegible]	63.328	25.523	24.224	32.910	.000	[illegible]
(40 , 5)	95.	87.863	117.728	120.216	117.889	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	30.338	25.276	[illegible]	[illegible]	[illegible]
(50 , 5)	155.	122.648	210.512	204.459	233.783	[illegible]	[illegible]
[illegible]	30.	111.143	.000	.000	.000	1.650	[illegible]
(41 , 6)	9.	14.232	10.455	14.716	14.683	[illegible]	[illegible]
[illegible]	3.	7.318	6.318	5.153	[illegible]	[illegible]	[illegible]
(104 , 6)	12.	13.664	16.325	14.391	14.851	15.177	[illegible]
(3 , 7)	24.	19.730	21.494	24.379	23.681	[illegible]	[illegible]
(25 , 7)	20.	21.422	21.794	25.120	24.214	[illegible]	[illegible]
(26 , 7)	33.	23.893	27.636	30.977	31.360	35.009	[illegible]
(50 , 7)	42.	28.485	35.051	42.026	44.489	45.280	[illegible]
(71 , 7)	30.	26.340	30.091	33.491	34.924	34.153	[illegible]
(73 , 7)	29.	21.354	24.992	23.580	22.687	19.526	[illegible]
(111 , 7)	22.	20.908	21.387	25.941	25.307	27.321	[illegible]
(44 , 8)	15.	13.684	14.642	14.998	14.975	15.908	[illegible]
(64 , 8)	5.	5.740	5.872	4.111	4.370	4.730	[illegible]
(104 , 8)	[illegible]	11.817	12.747	12.107	12.065	11.284	[illegible]
(107 , 9)	[illegible]	54.143	77.791	95.639	97.300	104.401	[illegible]
[illegible]	[illegible]	48.439	25.089	31.657	31.937	[illegible]	[illegible]
(91 , 10)	55.	54.924	70.213	62.044	[illegible]	59.252	[illegible]
[illegible]	[illegible]	2.941	2.258	2.206	[illegible]	[illegible]	[illegible]
[illegible]	3.	2.589	2.708	2.699	2.761	3.247	[illegible]
[illegible]	1.	1.256	1.247	1.184	1.200	[illegible]	[illegible]
(56 , 12)	[illegible]	1.048	1.046	1.017	1.100	[illegible]	[illegible]
[illegible]	[illegible]	1.248	1.341	1.348	1.264	1.147	[illegible]
(70 , 13)	[illegible]	51.892	50.136	53.913	[illegible]	52.659	[illegible]
[illegible]	[illegible]	216.186	243.604	41.548	[illegible]	[illegible]	[illegible]
[illegible]	380.	435.509	597.422	710.194	697.639	[illegible]	[illegible]

M	1				
[illegible]	.131E 00				
[illegible]	[illegible]				
[illegible]	63.53				
[illegible]	.129E 00	.356E-01			
[illegible]	.157E 05	.838E 04			
[illegible]	62.85	17.36			
[illegible]	[illegible]	.354E-01	.166E-01		
[illegible]	.157E 05	[illegible]	.497E 04		
[illegible]	62.35	17.32	8.11		
[illegible]	[illegible]	.352E-01	.164E-01	.129E-01	
[illegible]	[illegible]	[illegible]	.552E 04	[illegible]	
[illegible]	64.22	16.12	7.50	5.89	
[illegible]	.134E 00	.347E-01	.163E-01	.124E-01	[illegible]
[illegible]	.160E 05	.862E 04	.545E 04	.289E 04	.152E 04
[illegible]	63.43	16.42	7.71	5.85	2.88

CA 91 118x14 91 trous ε 10^{-4}
par tour avec remise à zéro

1	1)	310.	149.322 *	120.044 *	167.524 *	282.540 *	345.154 *	354.349 *
2	1)	[illegible]	[illegible] *	240.652 *	335.994 *	[illegible] *	[illegible] *	[illegible] *
3	2)	[illegible]	515.055 *	123.174 *	161.131 *	198.583 *	.000 *	.000 *
4	2)	350.	310.362 *	800.755 *	558.080 *	543.654 *	779.143 *	780.054 *
5	2)	170.	455.867 *	377.609 *	973.485 *	955.136 *	2221.095 *	1425.621 *
6	2)	110.	208.694 *	520.656 *	149.879 *	126.138 *	27.391 *	169.173 *
7	2)	310.	193.916 *	227.146 *	.000 *	.000 *	75.499 *	395.100 *
8	2)	320.	448.652 *	394.662 *	57.034 *	73.292 *	327.455 *	314.377 *
9	2)	500.	336.752 *	448.987 *	550.888 *	517.828 *	539.065 *	362.0.4 *
10	2)	280.	420.595 *	248.763 *	113.511 *	118.362 *	.000 *	.000 *
11	2)	200.	200.575 *	225.862 *	365.282 *	397.162 *	494.313 *	515.106 *
12	2)	410.	514.492 *	818.455 *	669.702 *	763.620 *	620.539 *	545.737 *
13	2)	130.	224.727 *	170.499 *	200.899 *	172.697 *	161.074 *	115.401 *
14	3)	550.	454.857 *	403.445 *	498.790 *	637.180 *	572.703 *	624.658 *
15	3)	705.	347.450 *	317.893 *	431.588 *	482.116 *	381.851 *	520.894 *
16	3)	340.	551.392 *	680.386 *	365.618 *	307.974 *	588.132 *	763.841 *
17	3)	60.	252.163 *	246.569 *	423.922 *	345.600 *	301.481 *	305.151 *
18	4)	55.	25.364 *	22.354 *	29.370 *	38.324 *	38.794 *	37.403 *
19	4)	45.	38.669 *	42.249 *	42.714 *	40.274 *	40.603 *	43.892 *
20	4)	38.	41.508 *	46.807 *	47.251 *	42.963 *	43.029 *	41.344 *
21	4)	6.	42.192 *	33.005 *	23.254 *	37.467 *	38.661 *	46.435 *
22	4)	51.	33.798 *	30.247 *	33.285 *	38.649 *	38.907 *	38.484 *
23	4)	41.	21.460 *	18.783 *	20.613 *	28.963 *	29.652 *	30.759 *
24	4)	33.	36.918 *	41.176 *	45.880 *	46.618 *	46.640 *	41.137 *
25	4)	43.	34.966 *	39.365 *	45.939 *	40.194 *	40.628 *	43.409 *
26	4)	47.	37.562 *	43.047 *	46.669 *	38.948 *	39.187 *	40.647 *
27	5)	20.	73.368 *	59.461 *	62.745 *	66.573 *	.000 *	.000 *
28	5)	20.	63.810 *	24.372 *	22.749 *	32.304 *	143.366 *	106.322 *
29	5)	45.	94.161 *	31.866 *	33.952 *	100.088 *	121.940 *	141.727 *
30	5)	95.	88.5[illegible] *	119.341 *	120.444 *	114.843 *	262.270 *	213.558 *
31	5)	70.	66.143 *	28.930 *	22.745 *	18.679 *	52.545 *	95.659 *
32	5)	155.	123.577 *	214.144 *	203.228 *	241.263 *	124.617 *	86.739 *
33	5)	20.	57.741 *	51.682 *	61.940 *	41.509 *	.000 *	.000 *
34	5)	30.	112.030 *	.000 *	.000 *	.000 *	2.206 *	12.342 *
35	5)	65.	84.232 *	67.215 *	52.654 *	64.171 *	77.147 *	46.690 *
36	6)	9.	14.269 *	10.177 *	15.022 *	15.104 *	14.958 *	15.327 *
37	6)	3.	7.841 *	6.248 *	5.032 *	4.291 *	3.468 *	4.949 *
38	6)	12.	13.651 *	16.525 *	14.326 *	15.018 *	13.502 *	16.467 *
39	6)	10.	13.101 *	11.083 *	17.197 *	17.083 *	17.717 *	14.169 *
40	7)	24.	19.643 *	21.347 *	24.407 *	23.518 *	27.924 *	27.117 *
41	7)	22.	21.148 *	23.812 *	22.565 *	23.010 *	31.566 *	25.983 *
42	7)	40.	27.500 *	29.028 *	31.214 *	32.844 *	37.057 *	40.454 *
43	7)	25.	19.540 *	22.303 *	26.014 *	25.363 *	28.406 *	42.430 *
44	7)	20.	21.314 *	21.703 *	25.057 *	24.045 *	26.097 *	22.485 *
45	7)	33.	23.768 *	27.356 *	31.046 *	31.096 *	37.092 *	35.486 *
46	7)	5.	16.571 *	15.304 *	10.465 *	8.506 *	6.211 *	6.138 *
47	7)	42.	26.379 *	34.687 *	42.205 *	44.879 *	48.220 *	48.159 *
48	7)	30.	26.008 *	29.814 *	33.633 *	35.277 *	33.795 *	32.467 *
49	7)	22.	21.259 *	24.826 *	23.648 *	23.104 *	17.599 *	19.581 *
50	7)	8.	12.244 *	10.062 *	8.346 *	6.575 *	3.625 *	4.655 *
51	7)	5.	8.745 *	7.698 *	4.738 *	3.994 *	3.980 *	5.147 *
52	7)	22.	20.815 *	21.313 *	25.923 *	25.183 *	28.408 *	27.870 *
53	8)	3.	10.352 *	11.032 *	10.483 *	10.462 *	11.062 *	11.112 *
54	8)	12.	11.193 *	11.873 *	10.420 *	10.175 *	11.420 *	11.086 *
55	8)	11.	11.070 *	9.535 *	9.188 *	9.542 *	8.897 *	8.861 *
56	8)	15.	13.595 *	14.687 *	15.089 *	15.086 *	16.131 *	15.500 *
57	8)	17.	11.034 *	9.485 *	9.664 *	9.622 *	9.064 *	9.817 *
58	8)	5.	5.702 *	5.792 *	4.305 *	4.466 *	4.699 *	6.304 *
59	8)	10.	11.740 *	12.814 *	12.304 *	12.153 *	11.007 *	11.291 *
60	9)	65.	51.224 *	42.419 *	51.445 *	49.447 *	54.693 *	53.487 *
61	9)	23.	46.131 *	16.074 *	18.259 *	19.556 *	18.197 *	18.806 *
62	9)	6.	30.536 *	27.887 *	11.699 *	6.987 *	11.184 *	12.839 *
63	9)	53.	64.285 *	77.674 *	94.557 *	95.399 *	106.964 *	103.410 *
64	10)	31.	48.067 *	25.694 *	31.211 *	31.513 *	33.072 *	38.735 *
65	10)	57.	59.121 *	61.530 *	62.719 *	65.089 *	66.020 *	67.346 *
66	[illegible])	33.	69.006 *	84.921 *	91.879 *	96.786 *	92.079 *	87.663 *
67	[illegible])	55.	54.502 *	6[illegible].844 *	62.079 *	63.344 *	57.637 *	57.748 *
68	11)	2.	2.502 *	1.640 *	1.648 *	1.552 *	1.511 *	1.489 *
69	11)	4.	2.794 *	2.792 *	2.777 *	2.948 *	3.050 *	3.015 *
70	11)	2.	2.921 *	2.210 *	2.177 *	2.170 *	2.205 *	2.172 *
71	[illegible])	1.	[illegible] *	2.[illegible] *	2.520 *	[illegible] *	[illegible] *	[illegible] *
72	1)	1.	2.590 *	1.466 *	1.467 *	1.150 *	1.034 *	.949 *
73	[illegible]	2.	2.532 *	2.791 *	2.832 *	2.743 *	[illegible] *	[illegible] *
74	1)	3.	2.672 *	2.680 *	2.675 *	2.807 *	3.2[illegible]7 *	3.283 *
75	1)	2.	1.527 *	1.709 *	1.6[illegible]7 *	2.405 *	2.169 *	2.179 *
76	1)	1.	1.516 *	1.368 *	1.644 *	1.791 *	1.690 *	1.709 *
77	2)	1.	1.232 *	1.217 *	1.148 *	1.195 *	1.207 *	1.200 *
78	2)	1.	1.094 *	1.084 *	1.055 *	1.102 *	1.124 *	1.227 *
79	[illegible]	3.	1.234 *	1.263 *	1.235 *	[illegible]	[illegible]	1.202
80	2)	1.	.902 *	1.222 *	1.144 *	1.376 *	1.431 *	1.443
81	2)	1.	1.027 -	1.025 -	.948 -	1.127 -	1.112 -	1.082

		0	1	2	3	4	5
[illegible]	1.	1.227	1.311	1.310	1.180	1.182	1.190
[illegible]	1.	.630	.643	.873	1.009	.984	[illegible]
	[illegible]	49.644	27.153	28.835	28.050	27.139	[illegible]
[illegible]	37.	54.783	63.168	63.514	62.531	57.428	[illegible]
[illegible]	70.	51.736	50.062	53.892	54.032	52.258	50.241
[illegible]	55.	169.050	180.818	237.532	274.521	536.169	[illegible]
[illegible]	65.	218.600	244.351	43.795	110.535	92.961	19.055
[illegible]	63.	394.434	463.526	269.982	238.901	253.328	325.102
	70.	540.019	274.716	342.102	302.336	305.024	430.886
	[illegible]	[illegible].467	598.491	704.466	[illegible].272	642.990	650.260

	1	2	3	4	5
[illegible]	.130E 00				
[illegible]	.141E 05				
[illegible]	65.56				
[illegible]	.131E 00	.355E-01			
[illegible]	.151E 05	.774F 04			
[illegible]CENTAGE	64.[illegible]	17.46			
[illegible]	.129E 00	.358E-01	.156E-01		
[illegible]	.153E 05	.788E 04	.465F 04		
[illegible]	63.61	17.63	7.68		
[illegible]	.143E 00	.367F-01	.166F-01	.128E-01	
[illegible]	.165E 05	.878F 04	.531F 04	.263E 04	
[illegible]	64.56	16.53	7.48	5.76	
[illegible]	.136E 00	.348E-01	.162E-01	.129E-01	.565E-02
[illegible]	.160E 05	.878E 04	.540E 04	.271E 04	.153E 04
[illegible]	63.63	16.39	7.62	6.05	2.66

CI 9 30x15 9 trous $\varepsilon = 10^{-4}$
par tour, négatifs

1 (11 , 1)	40.	10.155	24.337	28.253	28.138	28.458	28.626
2 (26 , 1)	1.	11.255	-3.468	-.315	-.457	.272	1.174
3 (22 , 3)	33.	29.314	37.991	37.985	41.453	41.294	25.395
4 (7 , 5)	13.	11.549	17.261	17.967	18.489	27.424	27.478
5 (21 , 5)	5.	11.788	5.506	10.040	10.543	10.067	10.256
6 (17 , 6)	43.	34.604	42.167	41.686	44.387	44.259	44.051
7 (14 , 8)	3.	19.496	-1.682	2.454	2.528	7.812	8.716
8 (20 , 8)	19.	18.231	27.896	29.142	29.086	28.339	28.072
9 (20 , 12)	21.	11.239	19.568	22.258	22.333	23.809	23.437
T		0	1	2	3	4	5

M	1	2	3	4	5
ALEUR PROPRE	.160E 00				
EST .293E 04	.839E 03				
POURCENTAGE	71.38				
ALEUR PROPRE	.176E 00	.333E-01			
EST .291E 04	.840E 03	.451E 03			
POURCENTAGE	71.16	13.34			
ALEUR PROPRE	.174E 00	.334E-01	.842E-02		
EST .291E 04	.840E 03	.452E 03	.353E 03		
POURCENTAGE	71.17	13.34	3.37		
ALEUR PROPRE	.177E 00	.335E-01	.855E-02	.785E-02	
EST .291E 04	.840E 03	.456E 03	.357E 03	.265E 03	
POURCENTAGE	70.88	13.45	3.43	3.15	
ALEUR PROPRE	.176E 00	.336E-01	.853E-02	.786E-02	.728E-02
EST .290E 04	.839E 03	.462E 03	.362E 03	.271E 03	.186E 03
POURCENTAGE	70.61	13.49	3.42	3.15	2.92

CI 23 30x15 23 trous . $\varepsilon = 10^{-4}$
par tour, négatifs

(11 , 1)	40.	10.458	24.505	28.035	28.789	28.133		29.86
(16 , 1)	0.2	1.156	.539	.717	1.393	1.384		1.72
(24 , 1)	3.	11.533	4.074	2.719	-4.544	-5.640		-11.65
(26 , 1)	1.	11.591	-3.373	-.395	-.139	-1.362		1.47
(6 , 2)	39.	42.362	37.651	47.119	47.001	43.502		43.29
(15 , 2)	49.	39.908	42.179	49.250	51.017	40.299		39.01
(17 , 3)	15.	31.102	18.566	19.574	41.738	38.315		32.31
(22 , 3)	33.	28.692	36.845	36.148	24.260	27.674		30.12
(8 , 4)	14.	15.086	22.293	25.376	29.970	21.115		23.16
(21 , 4)	44.	14.142	29.313	34.885	35.456	36.245		36.05
(22 , 4)	28.	14.371	21.559	19.937	23.412	25.331		24.39
(7 , 5)	13.	11.481	17.284	17.757	20.503	20.747		23.28
(21 , 5)	5.	10.833	5.592	9.707	9.532	9.749		8.95
(15 , 6)	34.	33.083	34.709	35.729	36.917	32.670		32.89
(17 , 6)	43.	36.058	42.545	42.548	42.464	44.557		46.21
(14 , 8)	3.	21.843	-1.897	-.199	-1.914	+1.217		3.63
(20 , 8)	19.	19.346	28.583	29.060	28.888	28.761		28.67
(2 , 10)	40.	33.247	35.709	33.406	31.869	31.643		73.12
(30 , 10)	42.	33.062	29.065	21.988	22.025	21.829		54.71
(14 , 11)	7.	53.089	8.143	-11.017	-14.330	-13.387		7.44
(20 , 12)	21.	11.874	20.031	22.488	22.613	23.063		23.59
(18 , 14)	46.	45.130	41.758	41.812	42.264	50.083		53.11
(20 , 14)	26.	46.560	40.482	27.509	27.704	27.463		23.48

M	1	2	3	4	5
UR PROPRE	.178E 00				
.248E 04	.814E 03				
RCENTAGE	71.70				
UR PROPRE	.183E 00	.352E-01			
.295E 04	.834E 03	.425E 03			
RCENTAGE	71.75	13.85			
UR PROPRE	.184E 00	.360E-01	.101E-01		
.300E 04	.864E 03	.446E 03	.329E 03		
RCENTAGE	71.17	13.93	3.90		
UR PROPRE	.184E 00	.354E-01	.960E-02	.731E-02	
.299E 04	.858E 03	.447E 03	.336E 03	.251E 03	
RCENTAGE	71.31	13.74	3.72	2.84	
UR PROPRE	.178E 00	.336E-01	.112E-01	.971E-02	.734E-02
.298E 04	.898E 03	.504E 03	.373E 03	.260E 03	.174E 03
RCENTAGE	69.82	13.23	4.40	3.82	2.89

CI 55 30x15 55 trous $\varepsilon = 10^{-4}$

par tour, négatifs

(1 , 1)	15.	11.818	24.864	24.928	25.396	26.364	35.910	26.097
(11 , 1)	45.	11.050	25.205	28.779	28.468	35.910	1.498	32.334
(16 , 1)	0.2	1.141	.549	.707	.640	1.498	-3.490	1.476
(24 , 1)	3.	11.241	4.007	2.692	1.554	-3.490	1.408	-3.118
(26 , 1)	1.	12.328	-3.526	-1.473	-3.130	1.408	[illegible]	-.947
(6 , 2)	33.	41.361	37.397	46.921	45.252	45.298	45.069	45.131
(15 , 2)	47.	39.586	41.636	49.075	52.936	45.069	43.501	52.294
(17 , 3)	15.	30.937	18.987	20.616	23.361	43.501	[illegible]	12.255
(20 , 3)	45.	24.693	42.221	44.729	44.586	44.239	[illegible]	49.351
(22 , 3)	33.	29.192	37.027	35.953	34.261	25.828	[illegible]	23.777
[illegible]	18.	16.286	23.153	25.234	27.559	19.821	[illegible]	17.228
(11 , 4)	2.	15.845	.744	5.860	4.817	17.506	[illegible]	18.584
(12 , 4)	4.	16.696	14.889	9.404	9.663	9.123	[illegible]	6.404
(21 , 4)	44.	15.071	29.948	34.119	34.260	36.853	[illegible]	36.380
(22 , 4)	28.	15.998	22.073	19.488	19.822	23.407	[illegible]	23.455
(7 , 5)	13.	11.522	17.035	17.470	20.973	19.931	[illegible]	19.226
(8 , 5)	11.	11.778	9.251	11.571	42.023	59.519	[illegible]	55.833
(10 , 5)	10.	12.273	6.956	9.878	29.446	45.757	[illegible]	45.183
(21 , 5)	8.	10.899	5.289	9.546	12.523	15.996	[illegible]	17.951
(12 , 6)	39.	36.282	39.037	37.841	38.050	37.921	[illegible]	40.939
(15 , 6)	34.	32.975	34.443	35.430	36.392	33.716	[illegible]	35.070
(17 , 6)	43.	36.842	41.847	42.449	41.600	44.837	[illegible]	27.171
(30 , 6)	28.	34.825	32.927	31.737	31.186	31.054	[illegible]	32.800
(8 , 7)	32.	36.307	37.844	36.486	19.442	20.519	[illegible]	20.151
(14 , 7)	23.	43.693	35.502	20.154	26.347	21.353	[illegible]	24.513
(19 , 7)	39.	35.322	35.660	38.872	41.391	40.125	[illegible]	40.693
(24 , 7)	41.	35.933	36.609	38.232	45.273	41.755	[illegible]	44.682
(27 , 7)	37.	35.518	34.171	33.841	35.030	33.770	[illegible]	34.992
(4 , 8)	12.	18.683	11.956	11.796	11.800	9.401	[illegible]	11.401
(6 , 8)	24.	18.219	20.732	19.681	19.835	19.610	[illegible]	19.597
(14 , 8)	3.	22.522	-.914	-1.155	1.350	-2.847	[illegible]	-11.245
(20 , 8)	19.	15.551	27.043	29.255	28.655	28.325	[illegible]	25.724
(28 , 8)	24.	17.256	26.338	25.490	25.457	25.155	[illegible]	22.146
(12 , 9)	8.	10.067	12.907	3.837	1.786	2.779	[illegible]	1.244
(2 , 10)	40.	36.644	38.060	35.105	35.486	34.786	[illegible]	37.349
(3 , 10)	23.	38.375	23.877	23.830	23.551	24.157	[illegible]	32.284
(10 , 10)	17.	39.217	18.323	14.539	16.808	18.402	[illegible]	11.453
(12 , 10)	33.	38.583	41.941	46.179	45.995	46.138	[illegible]	48.282
(14 , 10)	45.	45.291	76.670	57.163	58.018	57.423	[illegible]	83.851
(22 , 10)	21.	36.969	27.379	30.030	29.640	27.692	[illegible]	15.774
(26 , 10)	10.	40.851	10.652	7.269	6.318	4.824	[illegible]	-2.966
(30 , 10)	-2.	37.033	30.881	23.572	23.301	23.253	[illegible]	30.926
(1 , 11)	22.	49.056	22.044	20.471	20.127	19.564	[illegible]	19.403
(13 , 11)	47.	42.242	32.952	45.179	44.187	49.430	[illegible]	53.309
(14 , 11)	7.	56.737	10.182	-11.854	-6.608	-15.235	[illegible]	-35.135
(11 , 12)	1.	12.428	.579	5.019	2.647	7.074	[illegible]	.146
(17 , 12)	4.	13.298	5.657	9.179	7.150	7.241	[illegible]	-18.423
(20 , 12)	21.	10.615	18.968	22.815	21.360	20.388	[illegible]	27.124
(29 , 12)	1.	13.079	5.119	2.418	1.681	1.448	[illegible]	10.784
(13 , 14)	69.	39.758	41.580	68.565	68.939	72.325	[illegible]	72.986
(18 , 14)	46.	43.273	40.600	41.935	42.991	47.032	[illegible]	48.884
([illegible] , 14)	26.	36.870	38.561	26.898	28.760	30.829	[illegible]	30.601
(20 , 15)	71.	32.762	61.683	73.816	70.070	65.939	[illegible]	65.697
(23 , 15)	49.	37.613	49.640	59.280	57.168	53.008	[illegible]	50.087
(28 , 15)	54.	36.354	60.032	57.597	54.971	55.251	[illegible]	50.334
		0	1	2	3	4		5

M	1	2	3	4	5
[illegible]EUR PROPRE	.177E 00				
[illegible]ST .281E 04	.745E 03				
[illegible]OURCENTAGE	73.49				
[illegible]EUR PROPRE	.184E 00	.355E-01			
[illegible]T .296E 04	.813E 03	.399E 03			
[illegible]URCENTAGE	72.49	14.01			
[illegible]EUR PROPRE	.179E 00	.373E-01	.145E-01		
[illegible]ST .301E 04	.903E 03	.466E 03	.297E 03		
[illegible]URCENTAGE	69.36	14.53	5.64		
[illegible]EUR PROPRE	.178E 00	.422E-01	.210E-01	.999E-02	
[illegible]ST .316E 04	.107E 04	.569E 03	.323E 03	.205E 03	
[illegible]URCENTAGE	66.24	15.72	7.81	3.72	
[illegible]EUR PROPRE	.198E 00	.415E-01	.203E-01	.135E-01	.934E-02
[illegible]ST .331E 04	.113E 04	.648E 03	.411E 03	.253E 03	.144E 03
[illegible]URCENTAGE	67.09	14.08	6.89	4.59	3.17

CA 11 118x14 11 trous $\varepsilon = 10^{-4}$
par tour, négatifs

1 (54 , 2)	170.	457.591 *	364.916 *	993.488 *	1025.328 *	2551.838 *	1578.9[illegible]
2 (35 , 5)	20.	62.579 *	25.191 *	24.044 *	33.036 *	81.458 *	30.7[illegible]
3 (41 , 6)	9.	14.217 *	10.423 *	15.065 *	15.122 *	14.351 *	13.8[illegible]
4 (104 , 6)	12.	13.594 *	16.285 *	14.164 *	14.966 *	15.758 *	14.1[illegible]
5 (73 , 7)	29.	21.639 *	25.157 *	23.660 *	22.646 *	19.999 *	20.4[illegible]
6 (44 , 8)	15.	13.662 *	14.490 *	15.088 *	15.080 *	15.734 *	15.2[illegible]
7 (64 , 8)	5.	6.458 *	6.205 *	5.668 *	5.851 *	5.044 *	6.3[illegible]
8 (107 , 9)	83.	64.094 *	77.581 *	96.322 *	98.276 *	104.633 *	104.9[illegible]
9 (7 , 10)	31.	48.410 *	25.079 *	30.707 *	31.280 *	32.522 *	39.6[illegible]
0 (1 , 12)	1.	1.252 *	1.237 *	1.180 *	1.197 *	1.271 *	1.26[illegible]
1 (61 , 14)	65.	215.635 *	242.092 *	37.210 *	84.234 *	94.254 *	33.7[illegible]
T		0	1	2	3	4	5

M *	1 *	2 *	3 *	4 *	5
EUR PROPRE*	.126E 00 *	*	*	*	
T .415E 05*	.156E 05 *	*	*	*	
URCENTAGE *	62.40 *	*	*	*	
EUR PROPRE*	.126E 00 *	.350E-01 *	*	*	
T .418E 05*	.158E 05 *	.861E 04 *	*	*	
URCENTAGE *	62.11 *	17.27 *	*	*	
EUR PROPRE*	.126E 00 *	.346F-01 *	.162F-01 *	*	
T .417E 05*	.158E 05 *	.851F 04 *	.528F 04 *	*	
URCENTAGE *	62.25 *	17.11 *	7.99 *	*	
EUR PROPRE*	.137E 00 *	.359E-01 *	.163F-01 *	.130E-01 *	
T .449E 05*	.165E 05 *	.902E 04 *	.564F 04 *	.294E 04 *	
URCENTAGE *	63.34 *	16.59 *	7.52 *	5.99 *	
EUR PROPRE*	.129E 00 *	.356F-01 *	.162F-01 *	.113E-01 *	.617E-02
T .426F 05*	.163E 05 *	.865F 04 *	.531F 04 *	.297E 04 *	.170E 04
URCENTAGE *	62.42 *	17.27 *	7.84 *	5.49 *	2.99

CA 41 118x14 41 trous $\varepsilon = 10^{-4}$

par tour, négatifs

			0	1	2	3	4	5
1	(56 , 2)	360.	3 7.334	814.383	552.753	589.472	928.761	840.463
2	(5[illegible] , 2)	170.	4 5.923	354.006	983.568	1061.020	2411.659	1314.579
3	(64 , 2)	310.	1 3.952	235.390	-49.116	30.393	241.062	547.769
4	(109 , 2)	460.	4 6.061	178.229	841.671	865.154	237.016	171.098
5	(116 , 2)	190.	2 4.7[illegible]	170.226	198.873	256.095	102.609	-44.989
6	(12 , 3)	550.	4 6.350	408.375	491.289	701.322	664.505	765.490
7	(4 , 4)	56.	5.596	22.258	29.664	48.890	38.008	36.067
8	(31 , 4)	38.	1.855	47.196	47.166	43.554	44.632	43.559
9	(52 , 4)	6.	2.545	32.564	22.707	40.961	42.307	47.574
10	(60 , 4)	51.	4.[illegible]	30.154	33.342	38.916	3[illegible].308	37.372
11	(53 , 4)	41.	1.553	18.630	20.660	30.200	30.257	30.833
12	(20 , 5)	20.	2.[illegible]	59.027	61.898	64.746	-290.547	-402.259
13	(25 , 5)	20.	3.328	24.988	23.935	32.293	61.105	45.017
14	(40 , 5)	95.	7.803	117.853	119.947	117.954	147.192	172.846
15	(46 , 5)	70.	7.[illegible]	29.549	25.230	20.241	81.428	71.388
16	([illegible] , 5)	155.	1 2.043	211.149	205.418	234.581	225.526	289.292
17	([illegible] , 5)	30.	1 1.1[illegible]	-24.400	-26.572	-30.559	-18.653	-46.837
18	(41 , 6)	9.	4.282	10.463	14.662	14.709	14.574	13.671
19	(45 , 6)	3.	7.818	6.322	5.147	4.594	4.353	5.072
20	(124 , 6)	12.	3.664	16.331	14.456	14.893	14.343	16.052
21	(3 , 7)	24.	9.730	21.409	24.360	23.668	26.661	25.563
22	(25 , 7)	20.	1.422	21.753	25.110	24.213	25.870	21.366
23	(26 , 7)	33.	3.753	27.629	30.964	31.345	35.262	33.500
24	(50 , 7)	42.	6.465	35.060	42.048	44.510	45.177	45.009
25	(71 , 7)	30.	6.340	30.097	33.519	34.937	33.832	32.391
26	(73 , 7)	29.	1.354	24.999	23.586	22.689	19.151	20.601
27	(111 , 7)	22.	0.906	21.385	25.935	25.305	27.652	26.729
28	(44 , 8)	15.	3.654	14.642	14.997	14.977	15.977	15.470
29	(64 , 8)	5.	5.740	5.860	3.855	4.340	4.797	6.950
30	(104 , 8)	10.	1.817	12.749	12.115	12.076	10.609	11.260
31	(117 , 9)	83.	4.143	77.760	95.639	97.234	106.259	104.519
32	(7 , 10)	31.	8.439	25.160	31.865	32.078	33.451	39.900
33	(21 , 10)	5[illegible].	4.924	70.223	61.921	62.849	58.509	58.115
34	(41 , 11)	2.	2.941	2.260	2.197	2.211	2.283	2.230
35	(110 , 11)	3.	2.509	2.707	2.698	2.761	3.264	3.245
36	(1 , 12)	1.	1.256	1.247	1.184	1.201	1.254	1.237
37	(56 , 12)	1.	1.048	1.045	1.017	1.101	1.123	1.082
38	(96 , 12)	1.	1.248	1.341	1.348	1.264	1.239	1.284
39	(70 , 13)	70.	1.692	50.207	53.932	54.156	51.936	49.004
40	(51 , 14)	65.	2 6.166	243.380	41.650	88.078	76.888	5.529
41	(96 , 14)	360.	4 5.509	597.163	711.054	697.460	642.480	669.905
	Y		0	1	2	3	4	5

	1	2	3	4	5
LEUR PROPRE	.131E 00				
ST .422E 05	.152E 05				
OURCENTAGE	63.89				
LEUR PROPRE	.130E 00	.358E-01			
ST .426E 05	.158E 05	.840E 04			
OURCENTAGE	62.94	17.34			
LEUR PROPRE	.129E 00	.354E-01	.166E-01		
ST .425E 05	.157E 05	.840E 04	.497E 04		
OURCENTAGE	62.94	17.28	8.09		
LEUR PROPRE	.141E 00	.348E-01	.314E-01	.152E-01	.620E-02
ST .4[illegible]E 05	.197E 05	.125E 05	.601E 04	.286E 04	.158E 04
OURCENTAGE	59.64	14.75	13.30	6.45	2.63
LEUR PROPRE	.142E 00	.353E-01	.228E-01	.152E-01	
ST .476E 05	.180E 05	.107E 05	.597E 04	.281E 04	
OURCENTAGE	62.06	15.42	9.96	6.63	

CA 91 118x14 91 trous $\varepsilon = 10^{-4}$

par tour négatifs

1	(65 , 1)	310.	1 9.322	120.103	168.167	264.747	352.666	364.156
2	(14 , 2)	335.	1 1.195	245.865	332.307	356.862	1228.034	1046.384
3	(16 , 2)	160.	5 5.005	124.200	163.619	216.639	-347.938	-223.509
4	(36 , 2)	380.	3 0.302	797.209	558.700	510.880	802.657	868.470
5	(54 , 2)	170.	4 5.867	382.900	968.754	929.275	2190.108	1409.394
6	(55 , 2)	110.	2 8.694	517.828	147.743	92.978	40.921	225.875
7	(64 , 2)	310.	1 3.918	227.041	-24.743	-115.522	86.614	453.640
8	(79 , 2)	320.	4 8.652	369.324	55.650	86.167	230.323	224.954
9	(81 , 2)	600.	3 6.752	449.171	550.389	523.105	532.441	364.615
10	(84 , 2)	250.	4 0.595	248.142	115.266	124.054	-451.105	-305.648
11	(92 , 2)	500.	2 0.575	226.191	383.835	403.852	482.614	498.945
12	(109 , 2)	460.	5 4.492	824.761	878.467	785.882	652.848	620.916
13	(116 , 2)	120.	2 4.727	172.308	201.012	145.577	114.959	.847
14	(12 , 3)	550.	4 4.897	403.345	496.991	624.759	611.745	723.316
15	(29 , 3)	705.	3 7.450	317.657	430.673	480.636	402.601	564.155
16	(109 , 3)	340.	5 1.392	683.263	391.306	316.897	602.260	769.844
17	(118 , 3)	60.	2 2.163	246.629	424.513	361.672	300.038	293.625
18	(4 , 4)	58.	5.384	22.363	29.490	38.206	37.643	36.135
19	(23 , 4)	45.	6.659	42.232	42.744	40.071	41.864	45.631
20	(31 , 4)	38.	1.508	46.308	47.239	42.834	45.070	44.033
21	(57 , 4)	6.	2.192	33.044	23.096	35.563	37.630	47.320
22	(66 , 4)	51.	3.798	30.265	33.335	38.481	37.648	37.372
23	(83 , 4)	41.	1.460	18.797	20.651	28.424	29.170	30.503
24	(90 , 4)	33.	6.918	41.176	45.989	46.730	47.606	42.501
25	(93 , 4)	43.	4.986	39.374	46.063	39.836	43.523	47.492
26	(98 , 4)	47.	7.582	43.053	46.749	38.609	42.395	44.968
27	(20 , 5)	20.	3.368	59.104	62.165	65.966	-322.650	-432.748
28	(35 , 5)	20.	3.810	23.722	22.202	31.646	124.763	67.120
29	(37 , 5)	45.	4.161	30.874	32.662	104.896	157.428	195.753
30	(40 , 5)	95.	8.533	119.473	120.357	113.425	220.127	144.815
31	(46 , 5)	70.	6.143	27.975	22.174	14.219	31.395	42.624
32	(50 , 5)	155.	1 3.577	214.367	204.259	247.388	239.983	305.576
33	(69 , 5)	20.	7.741	51.479	61.187	36.415	27.621	61.998
34	(80 , 5)	30.	1 2.030	-28.987	-30.949	-36.326	-34.521	-48.415
35	(81 , 5)	64.	4.232	66.070	52.130	62.941	62.078	40.569
36	(41 , 6)	9.	4.203	10.189	14.997	14.899	14.610	15.213
37	(83 , 6)	3.	7.341	6.254	5.031	4.448	3.305	4.960
38	(104 , 6)	12.	3.651	16.531	14.368	15.110	14.091	16.084
39	(113 , 6)	10.	3.101	11.086	17.162	16.092	17.322	13.893
40	(3 , 7)	24.	9.643	21.342	24.332	23.436	27.172	26.304
41	(13 , 7)	27.	1.146	23.606	22.549	22.718	30.791	24.747
42	(16 , 7)	40.	7.300	29.028	31.234	32.783	38.926	40.149
43	(19 , 7)	28.	9.540	22.299	25.999	25.416	27.787	-3.122
44	(25 , 7)	20.	1.314	21.702	25.050	24.022	25.538	21.602
45	(26 , 7)	33.	3.708	27.330	31.033	30.876	36.379	34.677
46	(36 , 7)	5.	6.571	15.321	10.435	8.545	5.834	6.266
47	(50 , 7)	42.	5.379	34.701	42.226	44.855	45.993	46.257
48	(71 , 7)	30.	6.008	29.823	33.657	35.179	34.065	32.697
49	(73 , 7)	29.	1.253	24.836	23.656	23.329	16.166	20.327
50	(63 , 7)	8.	2.244	10.369	6.346	6.858	3.591	4.749
51	(45 , 7)	5.	6.745	7.702	4.735	4.103	3.652	5.162
52	(111 , 7)	22.	6.615	21.311	25.918	25.123	27.658	27.017
53	(10 , 8)	8.	0.332	11.032	10.443	10.453	11.758	11.145
54	(11 , 8)	12.	1.193	11.677	10.429	10.173	11.547	11.222
55	(37 , 8)	11.	1.070	9.533	9.187	9.606	8.354	3.453

56	(44 , 6)	15.	3.33[illegible]	14.687	15.090	15.074	16.200	15.584
57	(45 , 6)	17.	[illegible]	9.490	9.673	9.660	8.305	9.654
58	(64 , 8)	5.	[illegible]	5.795	4.195	3.940	4.561	6.161
59	(104 , 8)	10.	[illegible]	12.818	12.312	12.148	11.071	11.506
60	(2 , 9)	96.	[illegible]	42.431	51.522	49.474	55.184	54.366
61	(6 , 9)	36.	[illegible]	16.184	18.467	20.091	18.334	18.705
62	(60 , 9)	6.	[illegible]	27.492	11.570	6.753	11.642	13.810
63	(107 , 4)	63.	[illegible]	77.628	94.541	94.892	106.956	103.992
64	(7 , 10)	31.	[illegible]	25.775	31.380	31.845	33.116	38.747
65	(43 , 10)	67.	[illegible]	61.531	62.760	65.537	65.758	66.921
66	(71 , 10)	88.	9.006	84.932	91.911	96.710	91.925	87.465
67	(91 , 10)	56.	[illegible]	69.453	62.012	63.427	57.826	57.705
68	(6 , 11)	2.	[illegible]	1.644	1.649	1.545	1.480	1.444
69	(25 , 11)	4.	[illegible]	2.792	2.775	2.965	3.006	2.972
70	(41 , 11)	2.	2.921	2.213	2.174	2.172	2.178	2.131
71	(42 , 11)	5.	[illegible]	2.495	2.528	2.538	2.697	2.677
72	(53 , 11)	1.	[illegible]	1.472	1.460	1.124	1.029	1.001
73	(100 , 11)	2.	[illegible]	2.791	2.836	2.736	2.696	2.717
74	(110 , 11)	3.	2.972	2.679	2.674	2.828	3.206	3.139
75	(115 , 11)	2.	1.517	1.708	1.696	2.426	2.132	2.131
76	(118 , 11)	1.	[illegible]	1.368	1.648	1.808	1.656	1.636
77	(1 , 12)	1.	1.232	1.217	1.147	1.213	1.201	1.189
78	(17 , 12)	1.	[illegible]	1.084	1.055	1.116	1.110	1.207
79	(26 , 12)	3.	[illegible]	1.239	1.234	1.262	1.218	1.179
80	(36 , 12)	1.	[illegible]	1.220	1.148	1.364	1.407	1.465
81	(56 , 12)	1.	[illegible]	1.025	.998	1.145	1.063	1.041
82	(96 , 12)	1.	1.[illegible]	1.310	1.309	1.165	1.245	1.273
83	(118 , 12)	1.	[illegible]	.643	.877	1.015	.961	.946
84	(6 , 13)	40.	[illegible]	27.263	28.974	28.175	26.433	26.086
85	(27 , 13)	57.	[illegible]	63.169	63.521	62.392	54.465	56.679
86	(70 , 13)	70.	1.796	50.086	53.917	54.080	51.691	49.565
87	(14 , 14)	65.	1 [illegible]	180.966	235.759	273.165	503.596	352.194
88	(61 , 14)	65.	2 [illegible]	244.094	43.621	121.077	89.751	7.555
89	(26 , 14)	350.	3 [illegible]	463.323	271.723	238.953	261.601	342.655
90	(88 , 14)	170.	5 [illegible]	276.530	344.032	298.278	304.168	381.375
91	(96 , 14)	300.	4 [illegible]	596.124	704.976	673.762	653.543	678.619
	T		0	1	2	3	4	5

1 2 3 4 5

VALFL - PREPR.	.190E 06				
TEST .410E 05	.141E 05				
PERCENTAGE	55.[illegible]				
VALFL PREPR.	.131E [illegible]	.356E-01			
TEST .422E 05	.151E 05	.775F 04			
PERCENTAGE	64.[illegible]	17.41			
VALFL PREPR.	.131E [illegible]	.353E-01	.159F-01		
TEST .426E 05	.134E [illegible]	.794F 04	.466F 04		
PERCENTAGE	63.7[illegible]	17.60	7.73		
VALFL PREPR.	.150E [illegible]	.376F-01	.251F-01	.148E-01	
TEST .728E 05	.188E 05	.111E 05	.564E 04	.256E 04	
PERCENTAGE	64.5[illegible]	14.55	10.28	5.83	
VALFL PREPR.	.154E [illegible]	.334F-01	.340F-01	.132F-01	.562F-02
TEST .564E 05	.204E 05	.129F 05	.581F 04	.265E 04	.146E 04
PERCENTAGE	61.11	14.23	13.57	6.03	2.23

Pour la comparaison des analyses, nous avons effectués :

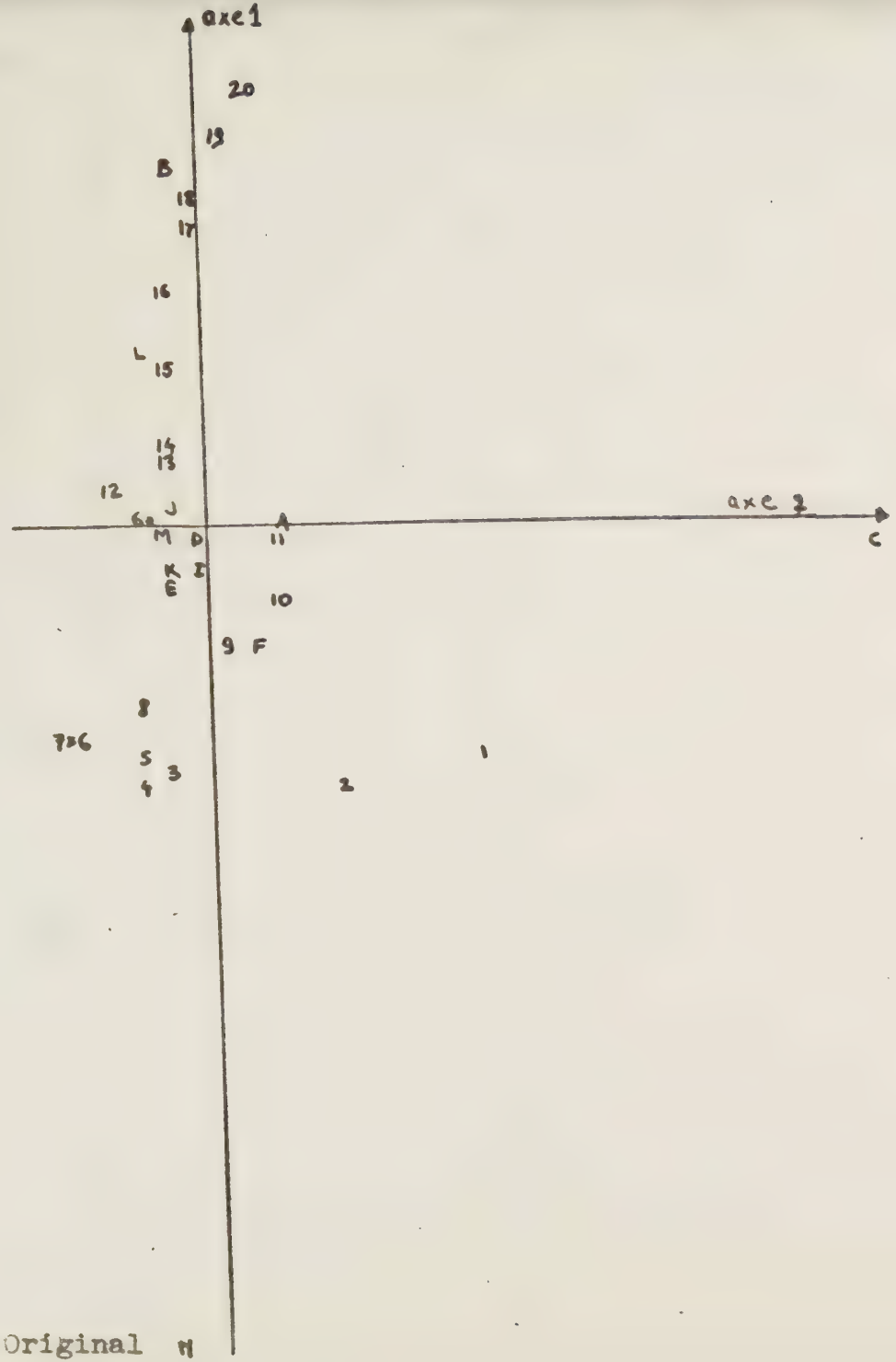
- l'analyse du tableau complet
- l'analyse du tableau, les trous remplacés par des 0
- l'analyse du tableau complété par le calcul.

Pour chaque cas, deux approximations de rang différent ont été retenues (Λ_1, Λ_0 ou

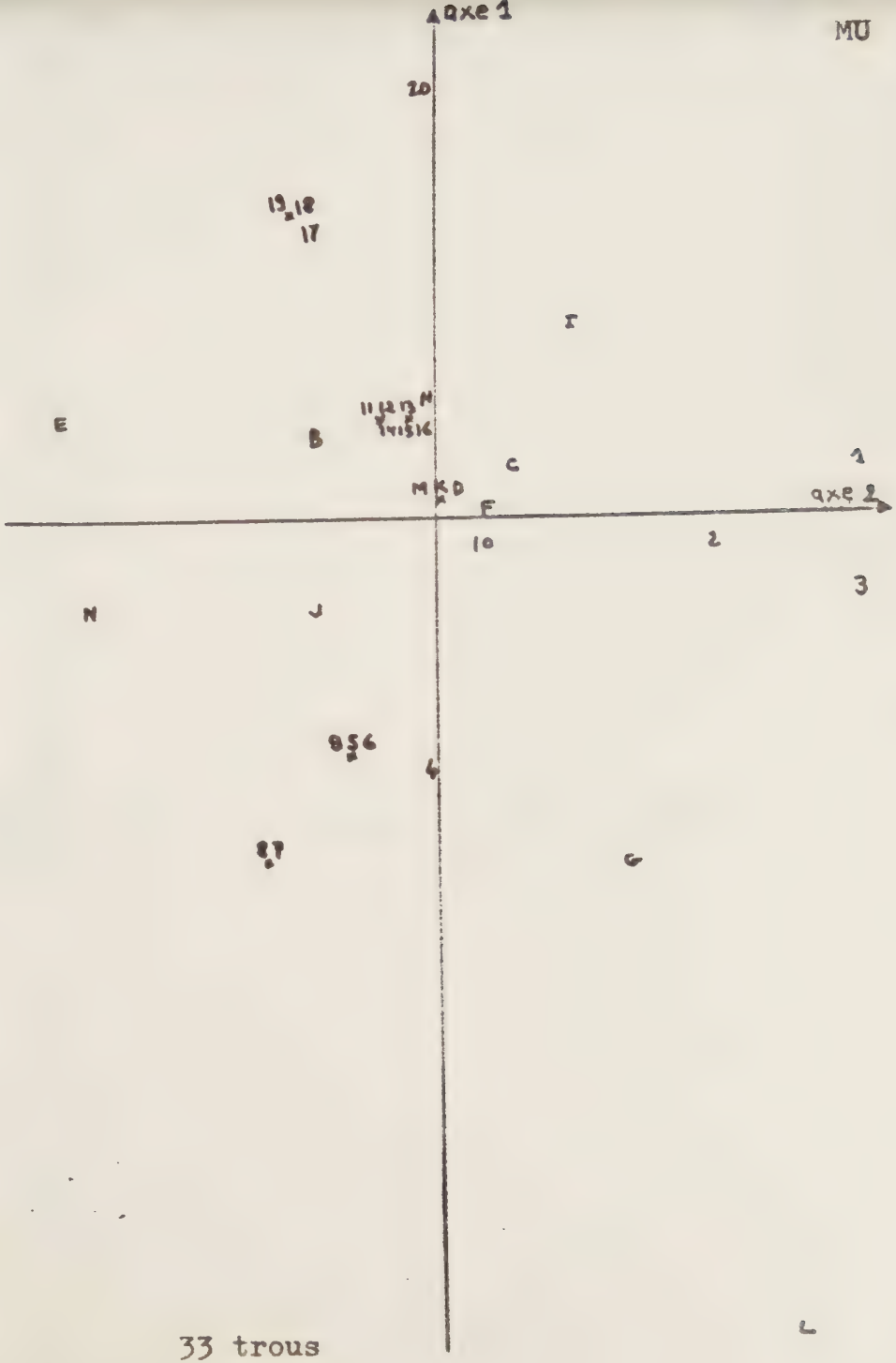
	rang	
MU	1	0
CI9	1	2
CI23	1	2
CI55	1	2
CA11	1	2
CA41	1	0
CA91	1	0

Ces approximations ont été retenues car l'étude des tableaux précédents montr
qu'en général l'approximation de rang 1 semble la mieux rapprochée du tableau
initial. Mais il faut remarquer que dans tous ce cas, la première valeur prop
est grande par rapport aux autres (77,71% ou 62 % de l'inertie totale).

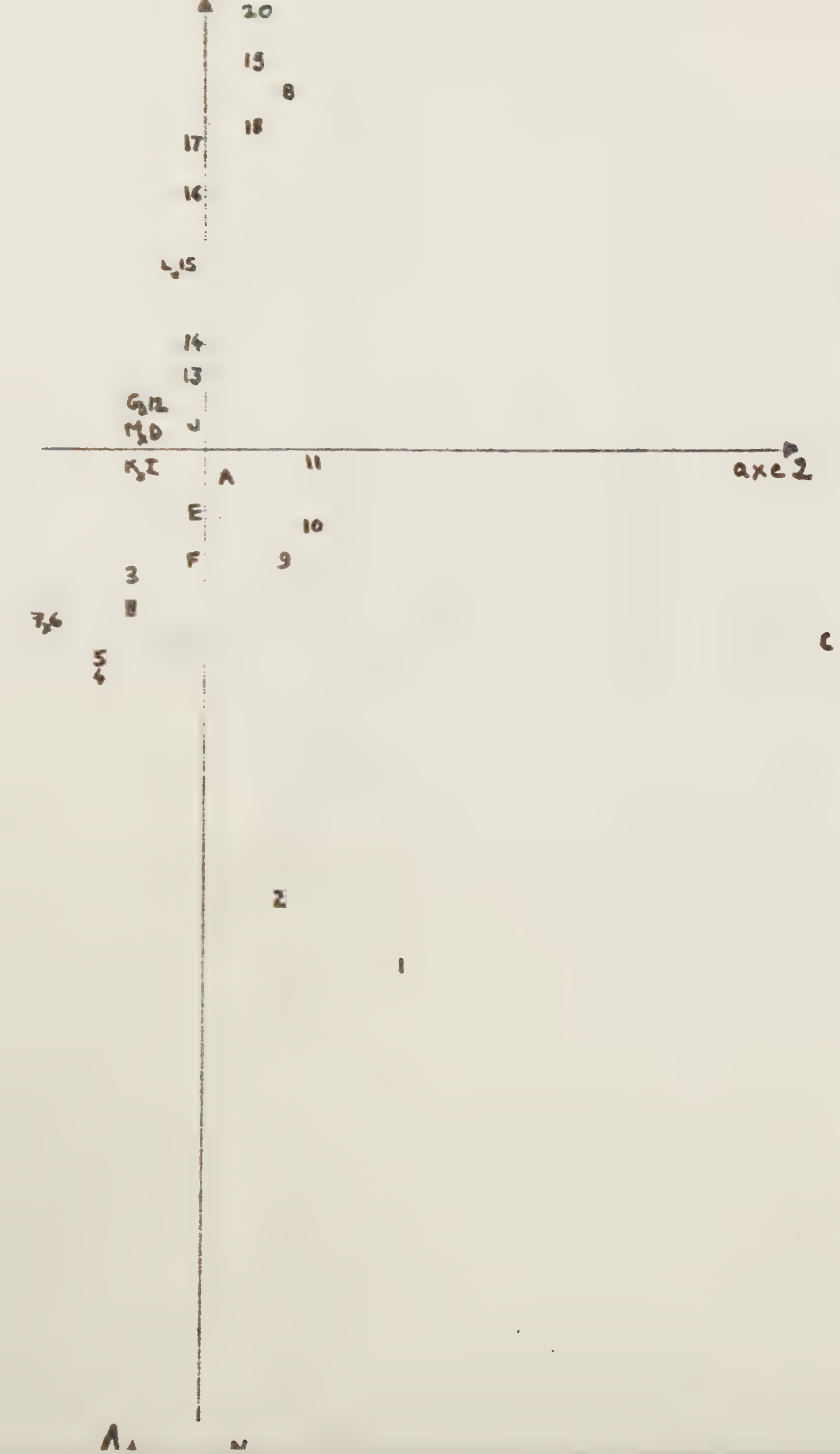
Les plans correspondants aux 3 premiers axes principaux d'inertie ont été tra
pour ces différentes analyses.



Original

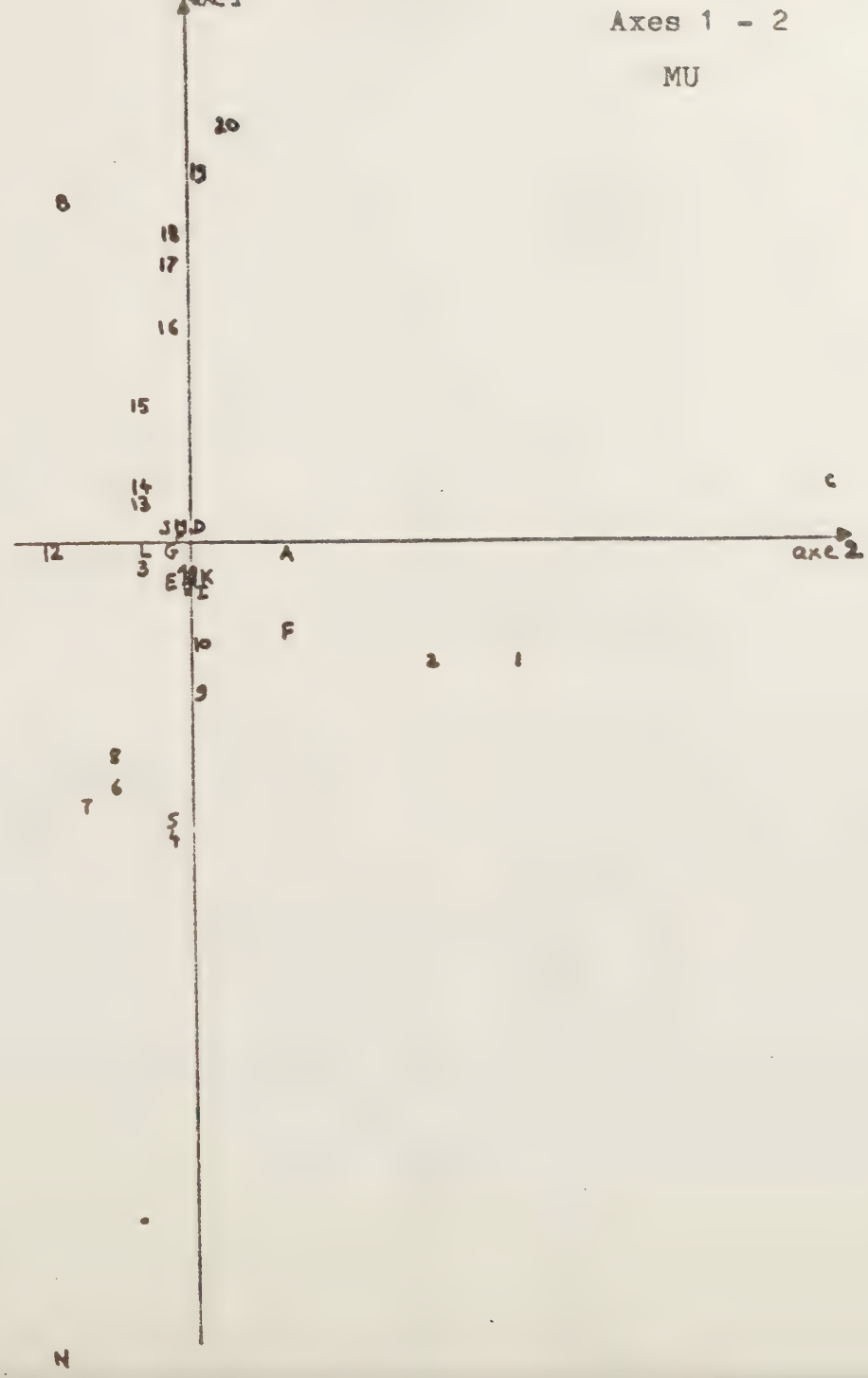


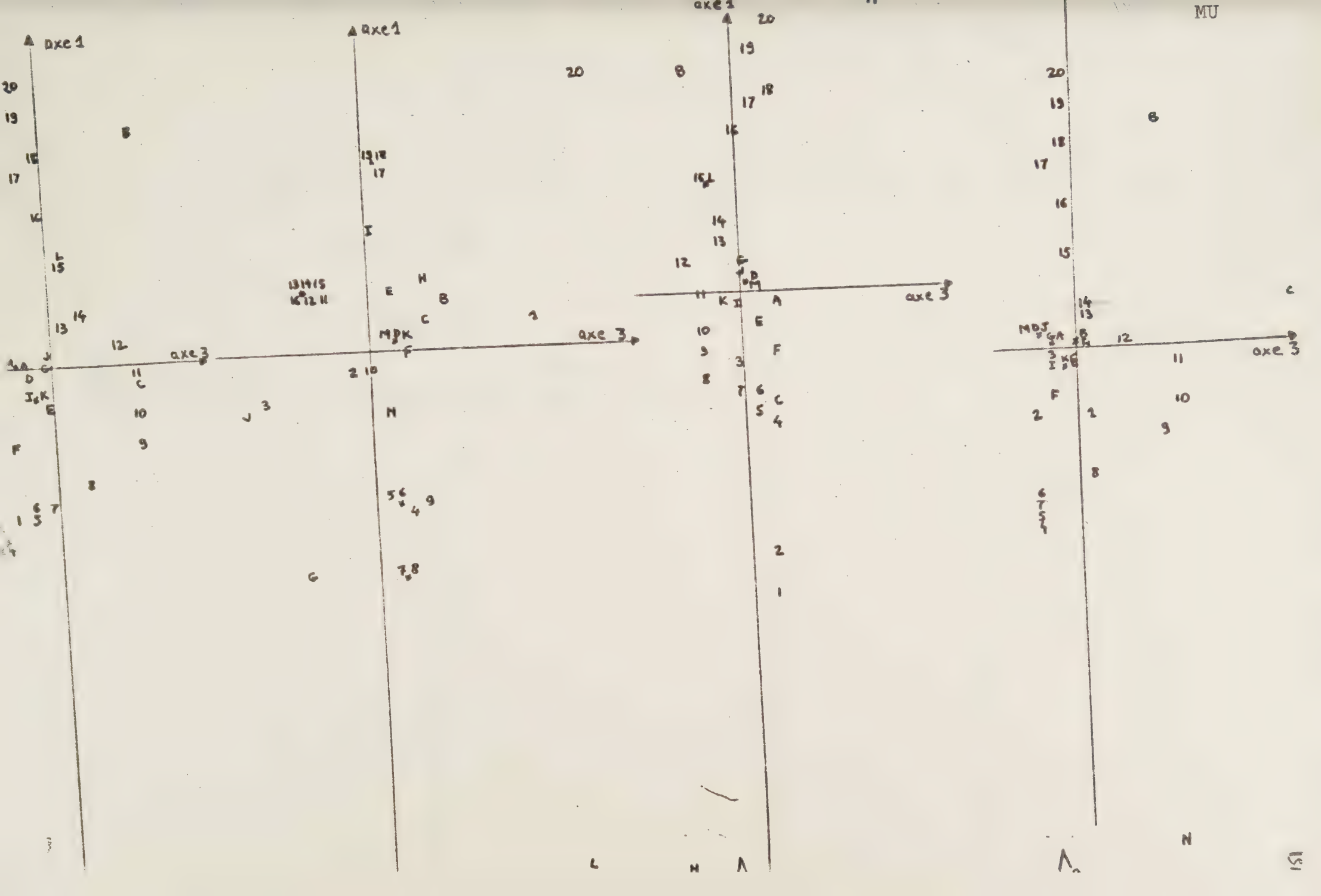
33 trous

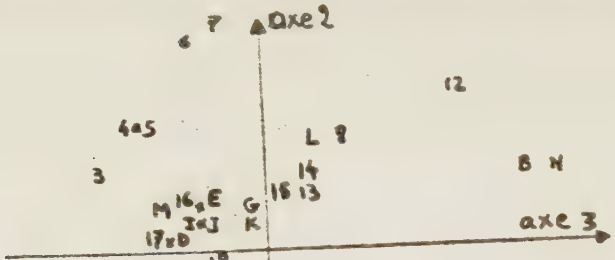

20
15
8
18
17
16
15
14
13
G,12
M,D
J
K,I
A
11
axe 2
E
10
F
9
3
B
3,6
5
4
C
2
1
A
N

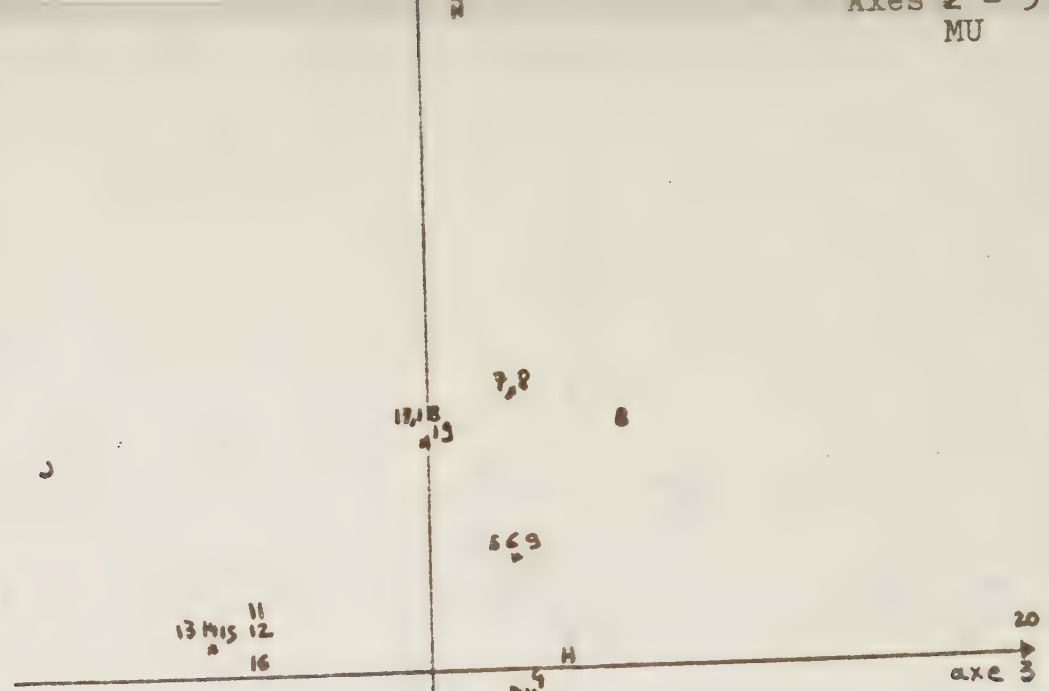
Axes 1 - 2

MU


20
8
18
17
16
15
14
13
J M D
12
L
3
G
A
axe 2
E B K
C
F
10
9
2
1
8
6
7
5
4
A
N

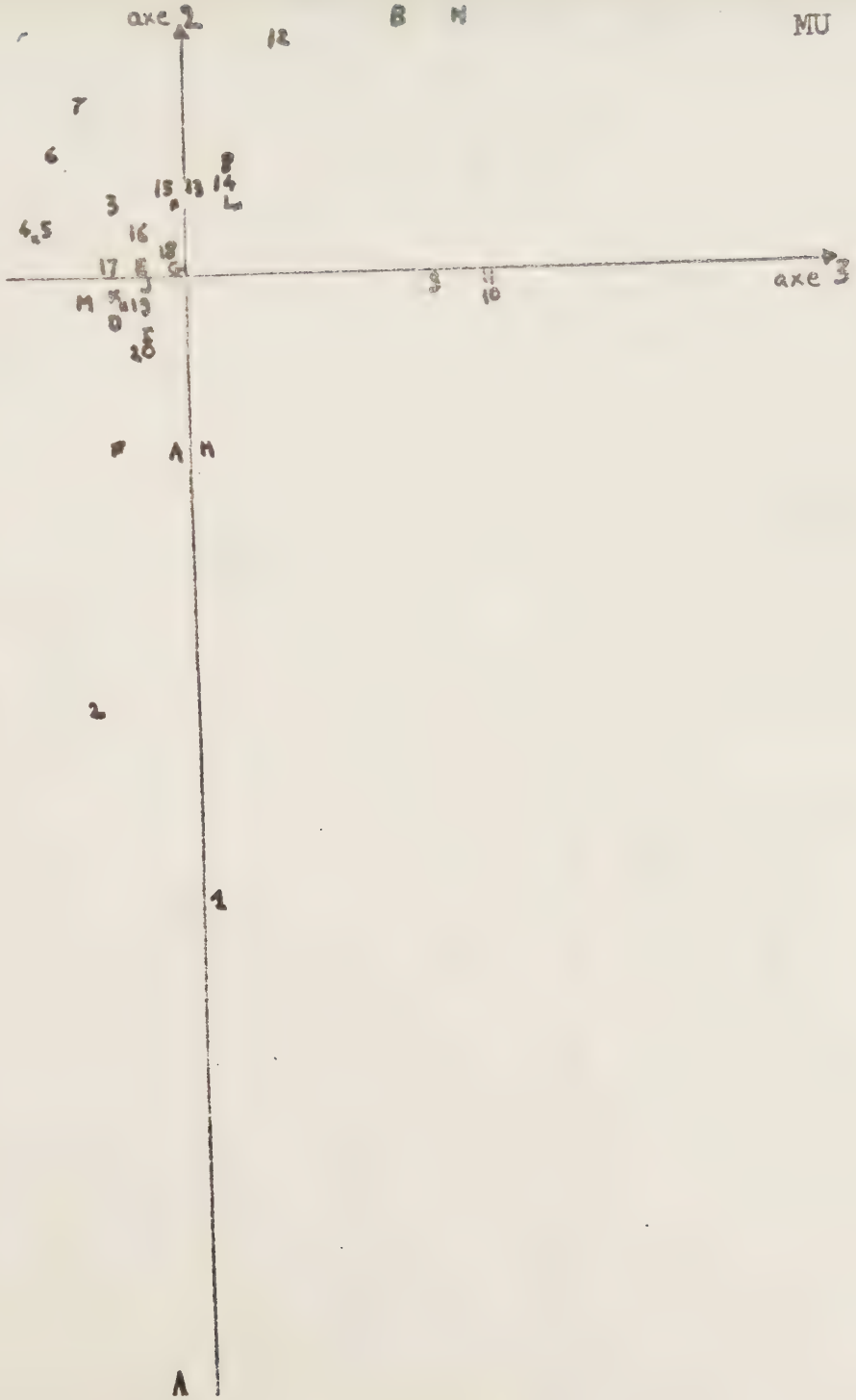

axe 1
axe 3
axe 1
axe 3
axe 1
axe 3
MU

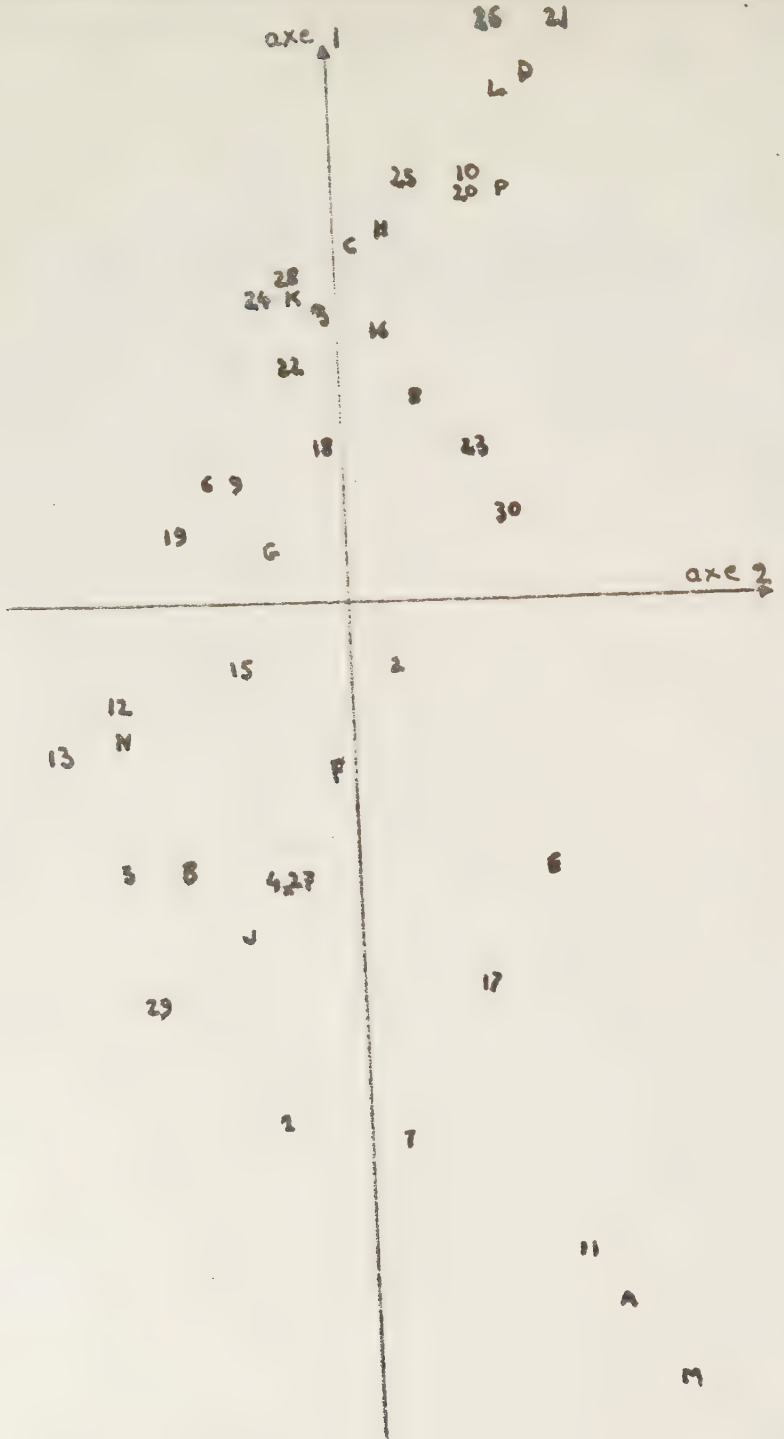

Axe 2
axe 3

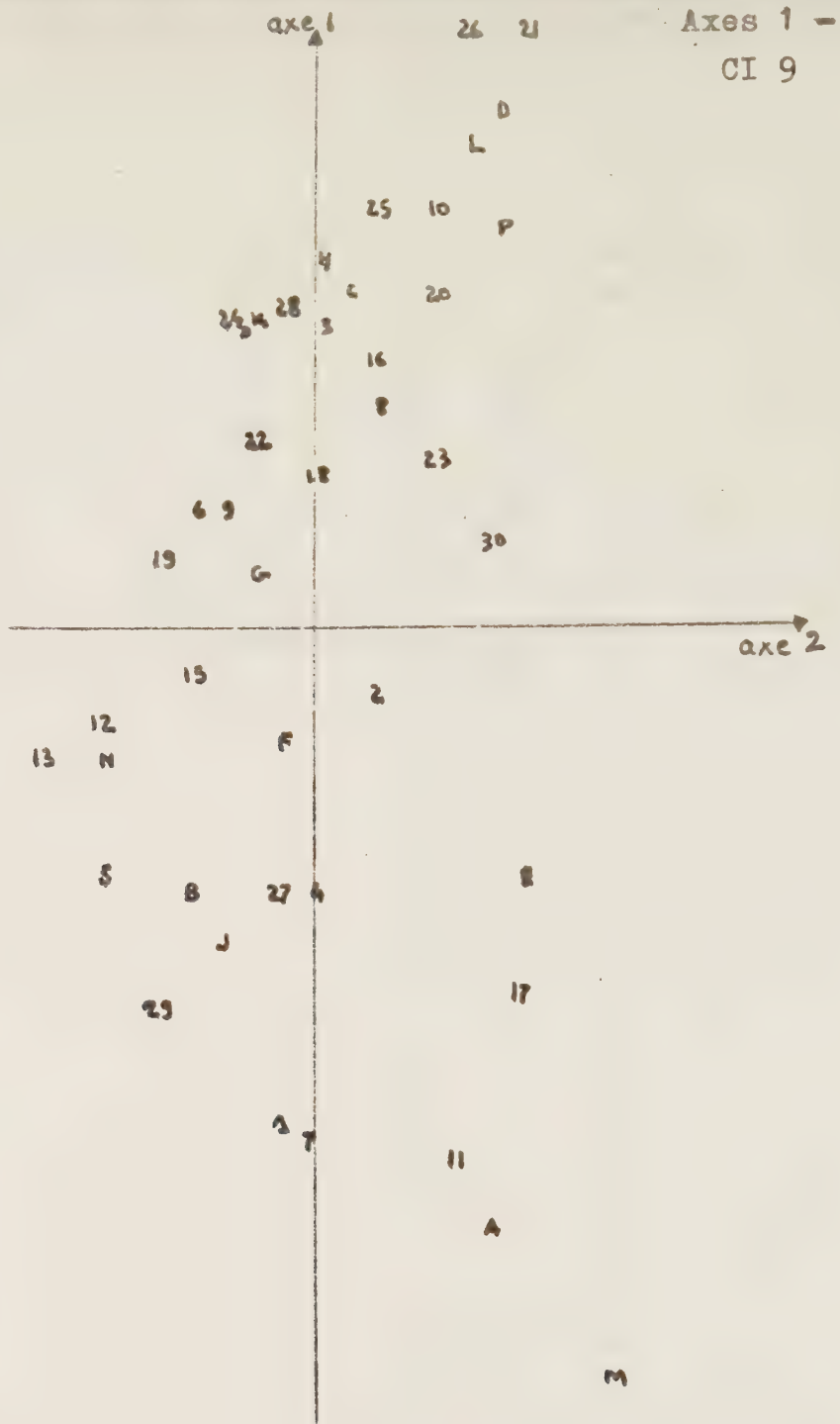

Axes 2 - 3
MU
axe 3

Axes 2-3

MU



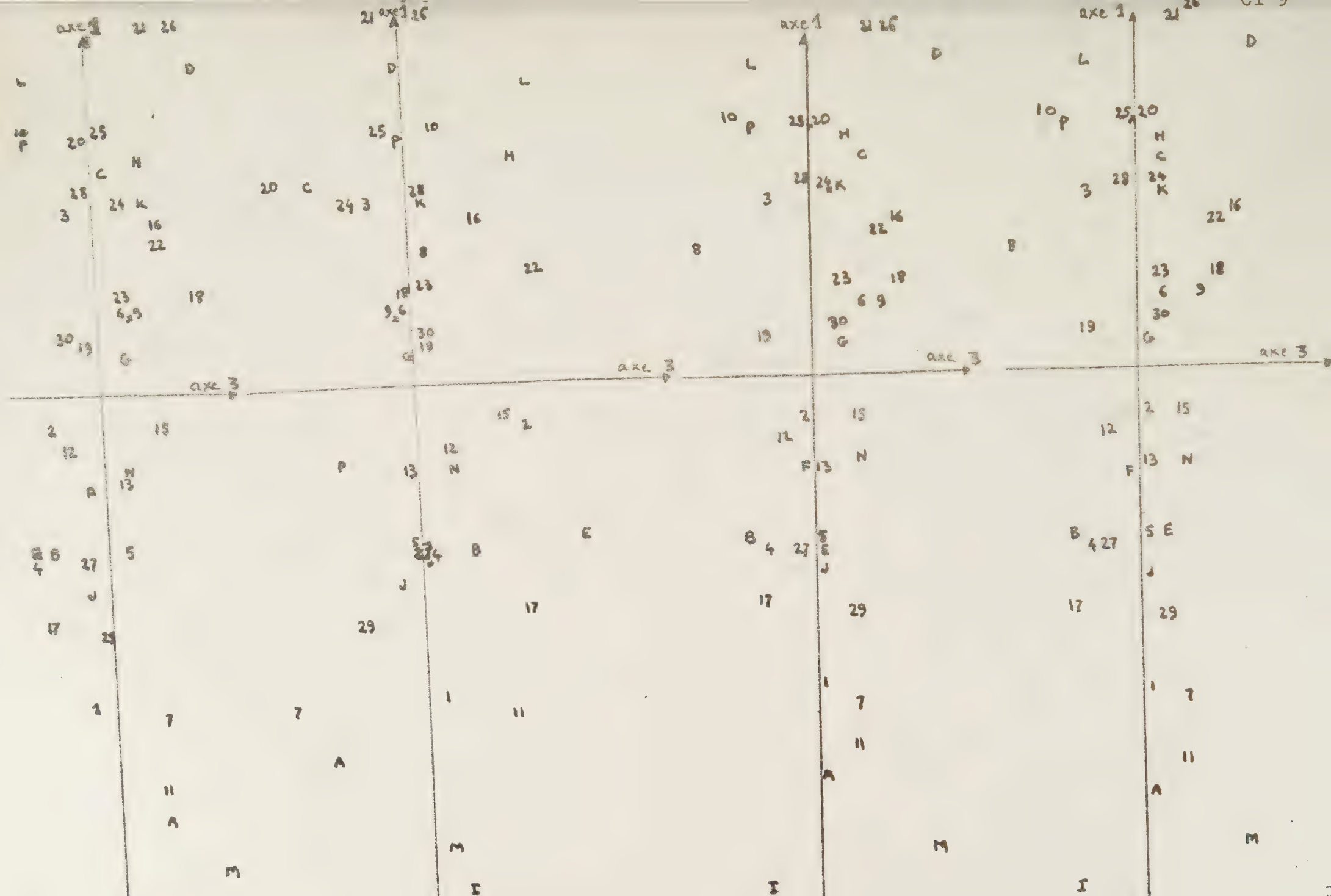

axe 1
axe 2
26
21
D
L
25
10
20
P
H
C
28
24
K
3
16
22
8
18
23
6
9
30
19
G
15
2
12
N
13
F
5
B
4
27
E
J
17
29
1
7
11
A
M
I

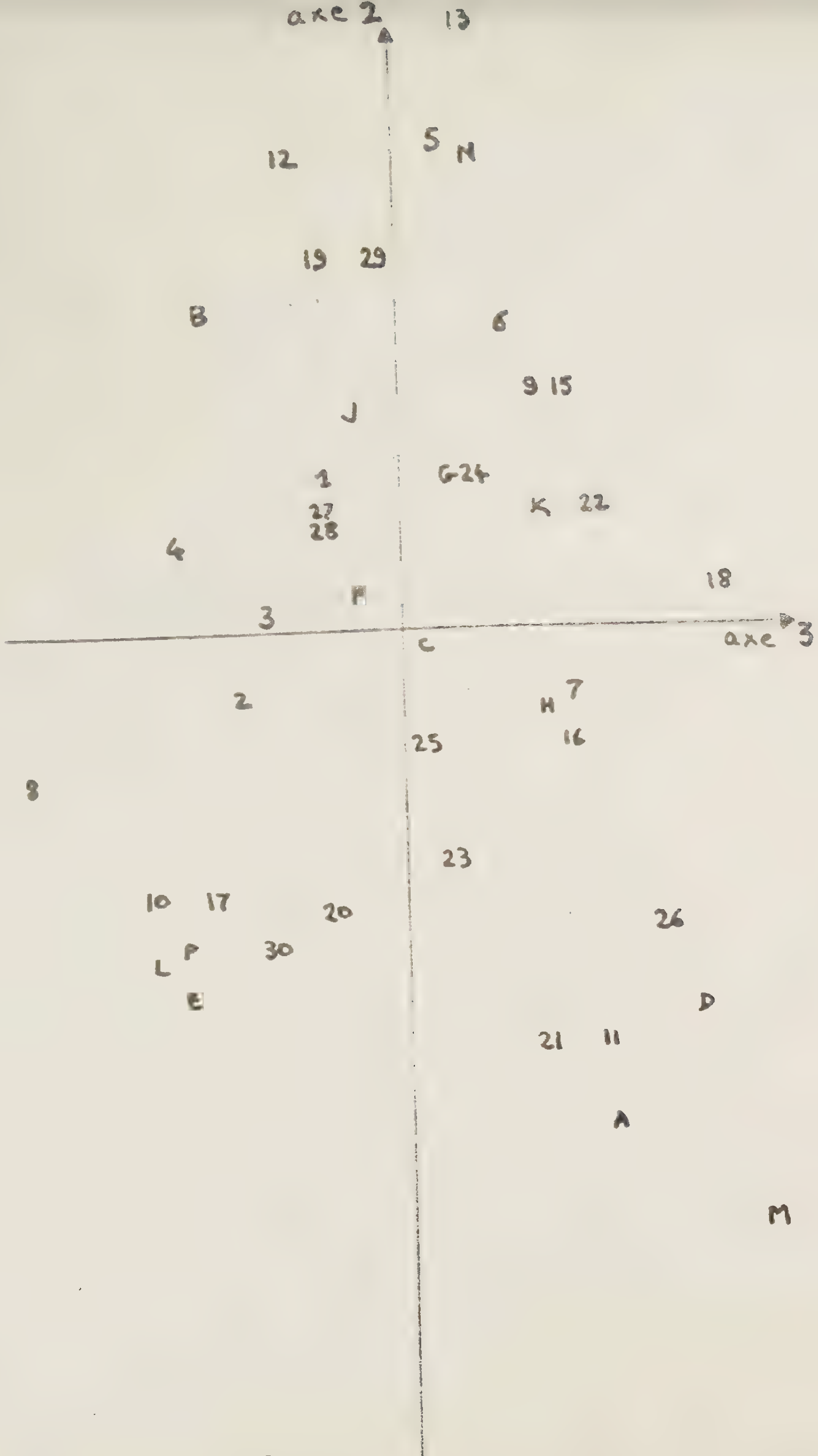

axe 1
axe 2
26
21
D
L
25
10
P
H
C
20
2
4
28
3
16
8
22
23
18
6
9
30
19
G
15
2
12
N
13
F
5
B
27
4
E
J
17
29
1
7
11
A
M

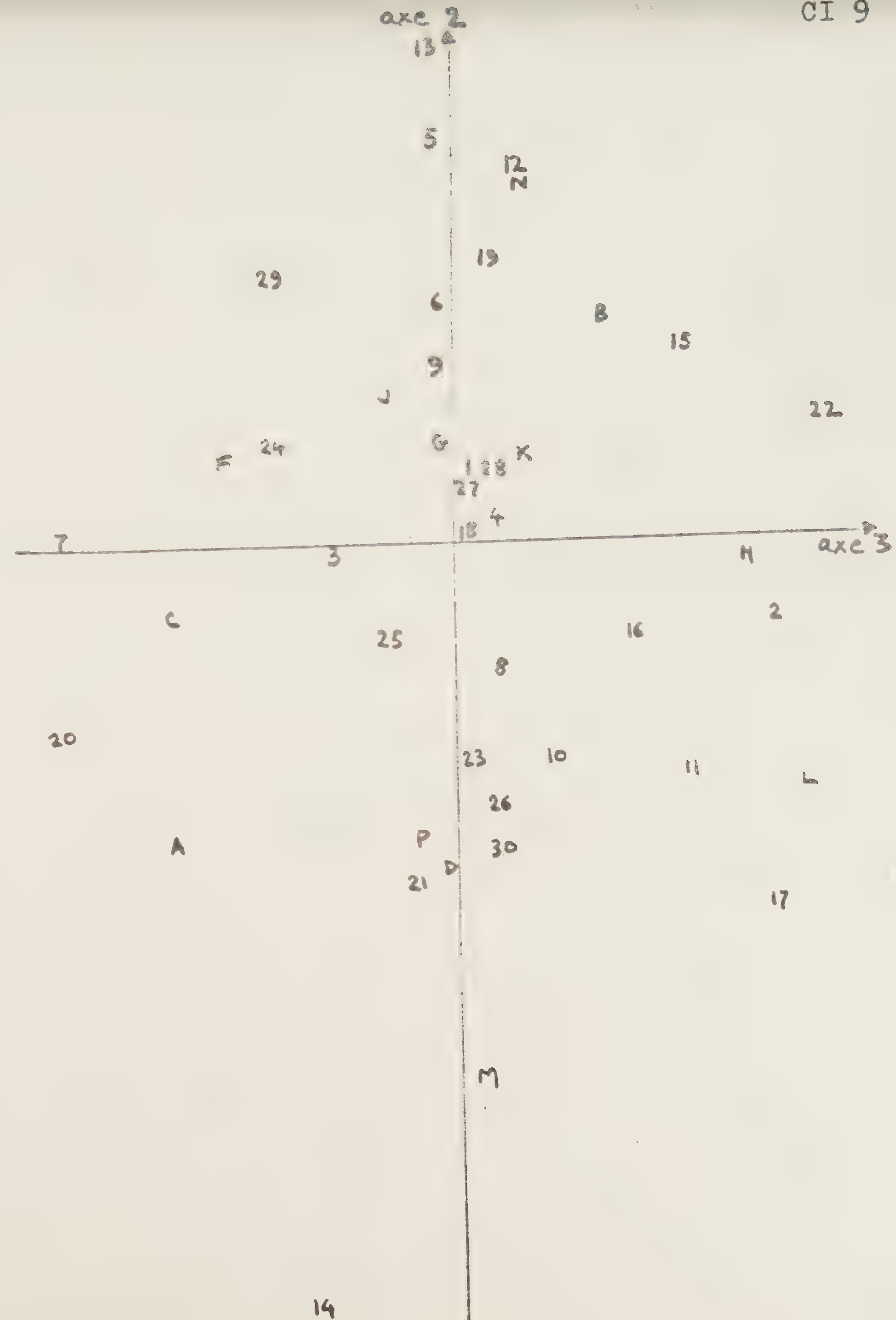
Axes 1 – 2
CI 9

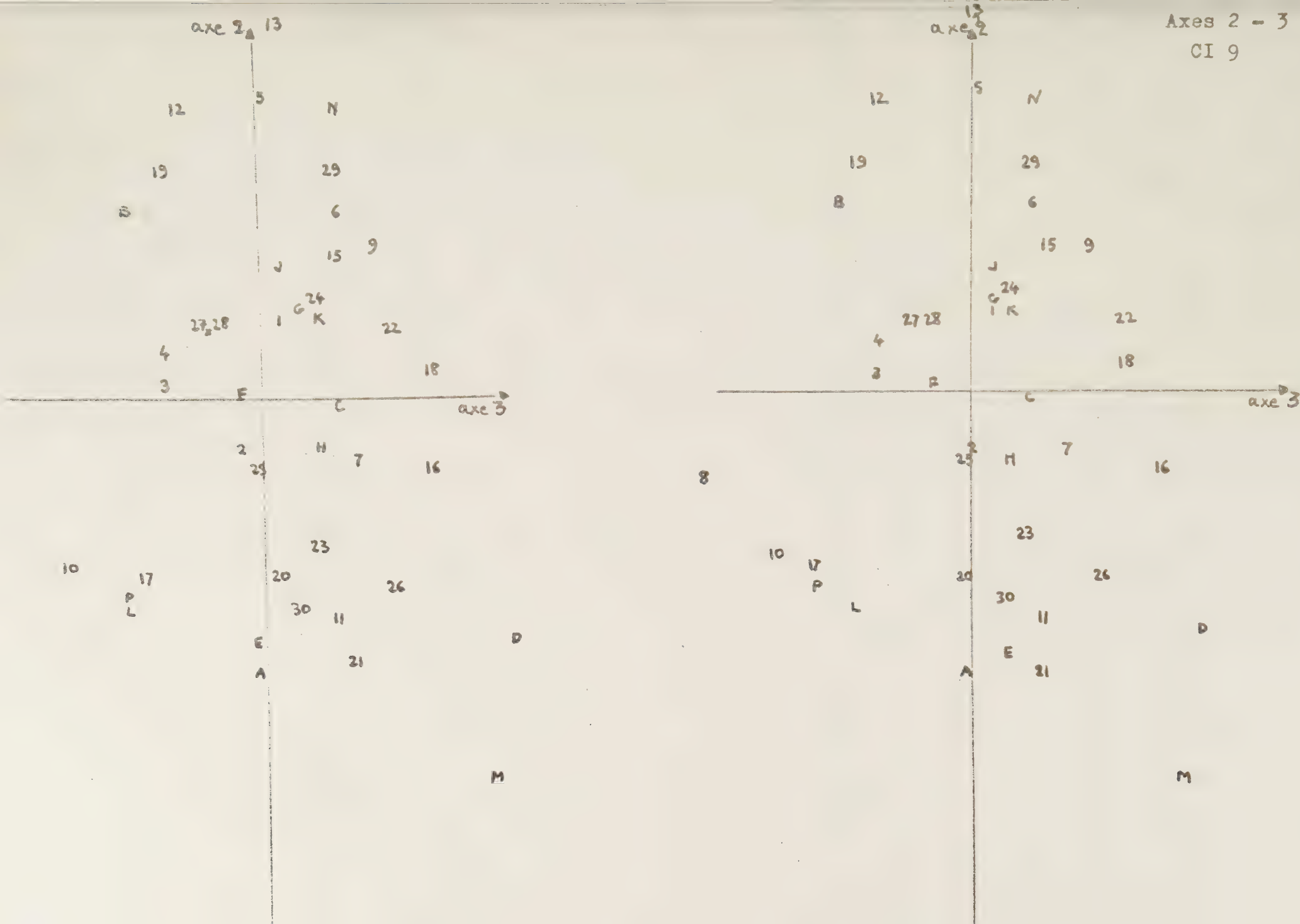
axe 1
axe 2

axe 1
axe 2


axe 1
axe 3
axe 1
axe 3
axe 1
axe 3

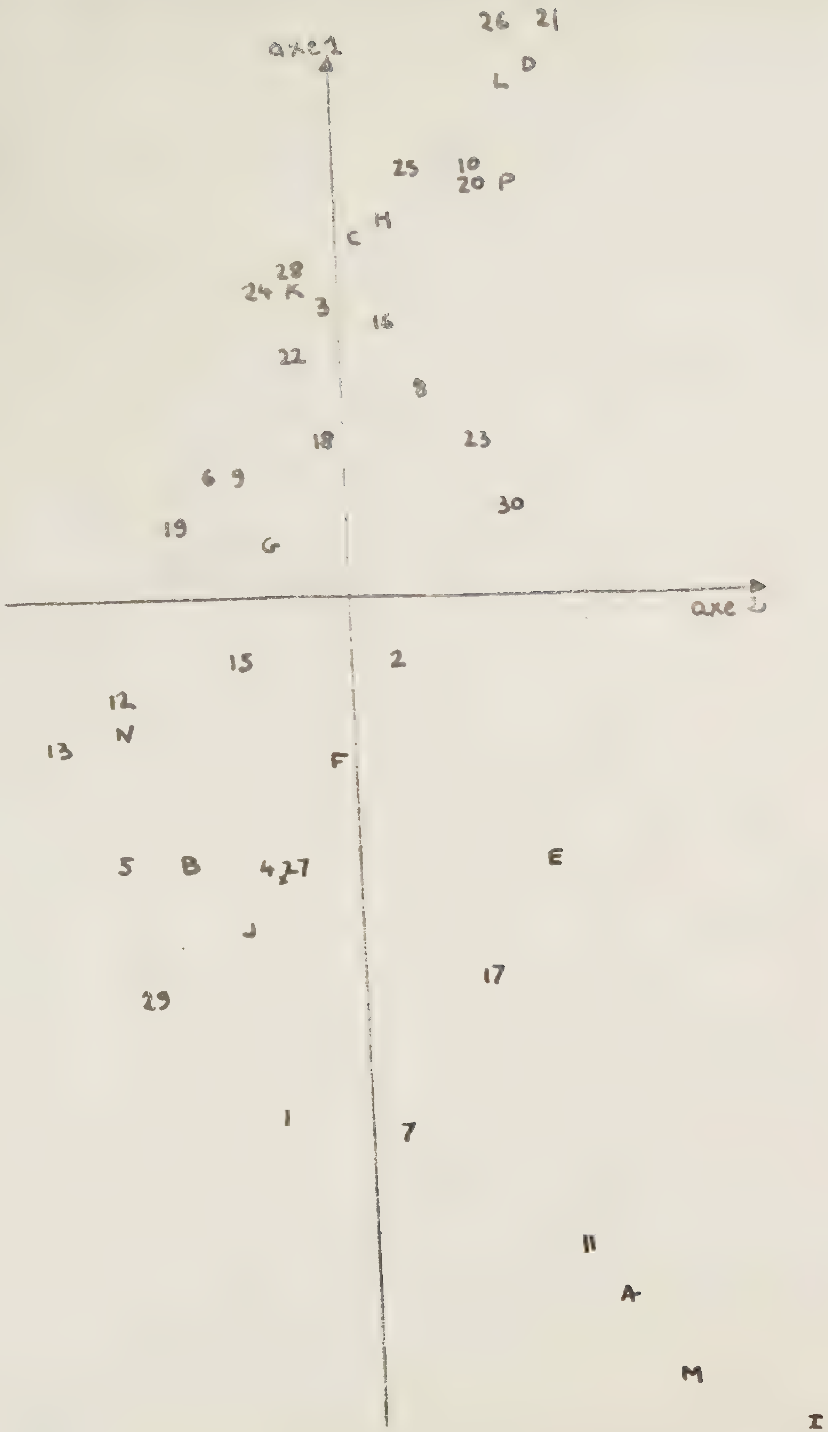

axe 2
axe 3

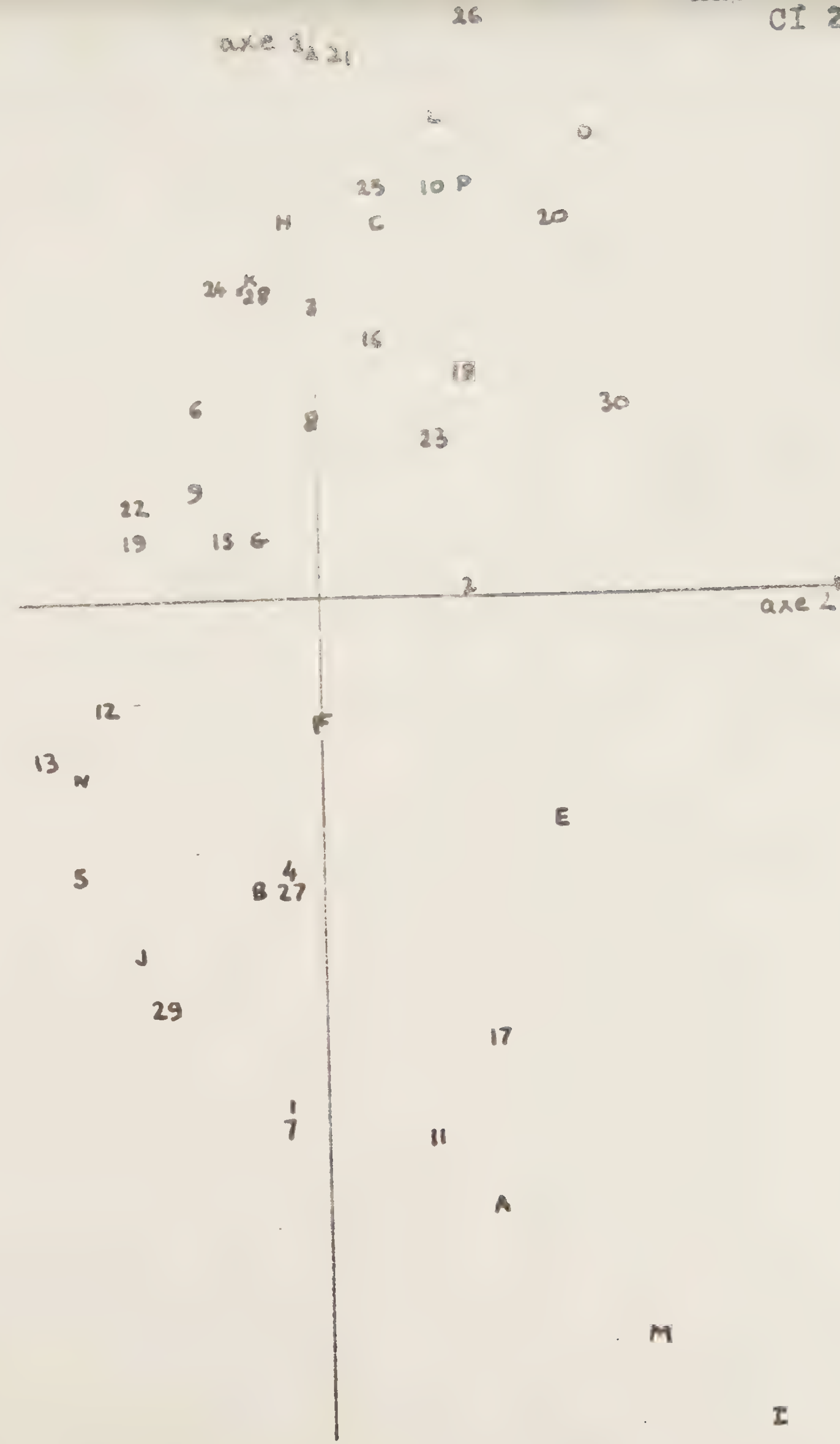

axe 2
axe 3


axe 2
axe 3
13
5
N
12
19
29
B
6
9
15
J
G
24
K
I
27,28
22
4
18
3
E
C
2
H
7
25
16
23
10
17
20
26
P
L
30
11
E
D
21
A
M
axe 2
axe 3
13
5
N
12
19
29
B
6
9
15
J
G
24
I
K
27
28
22
4
18
3
B
C
2
H
7
25
16
8
23
10
17
20
26
P
L
30
11
E
D
21
A
M

Axes 2 - 3
CI 9

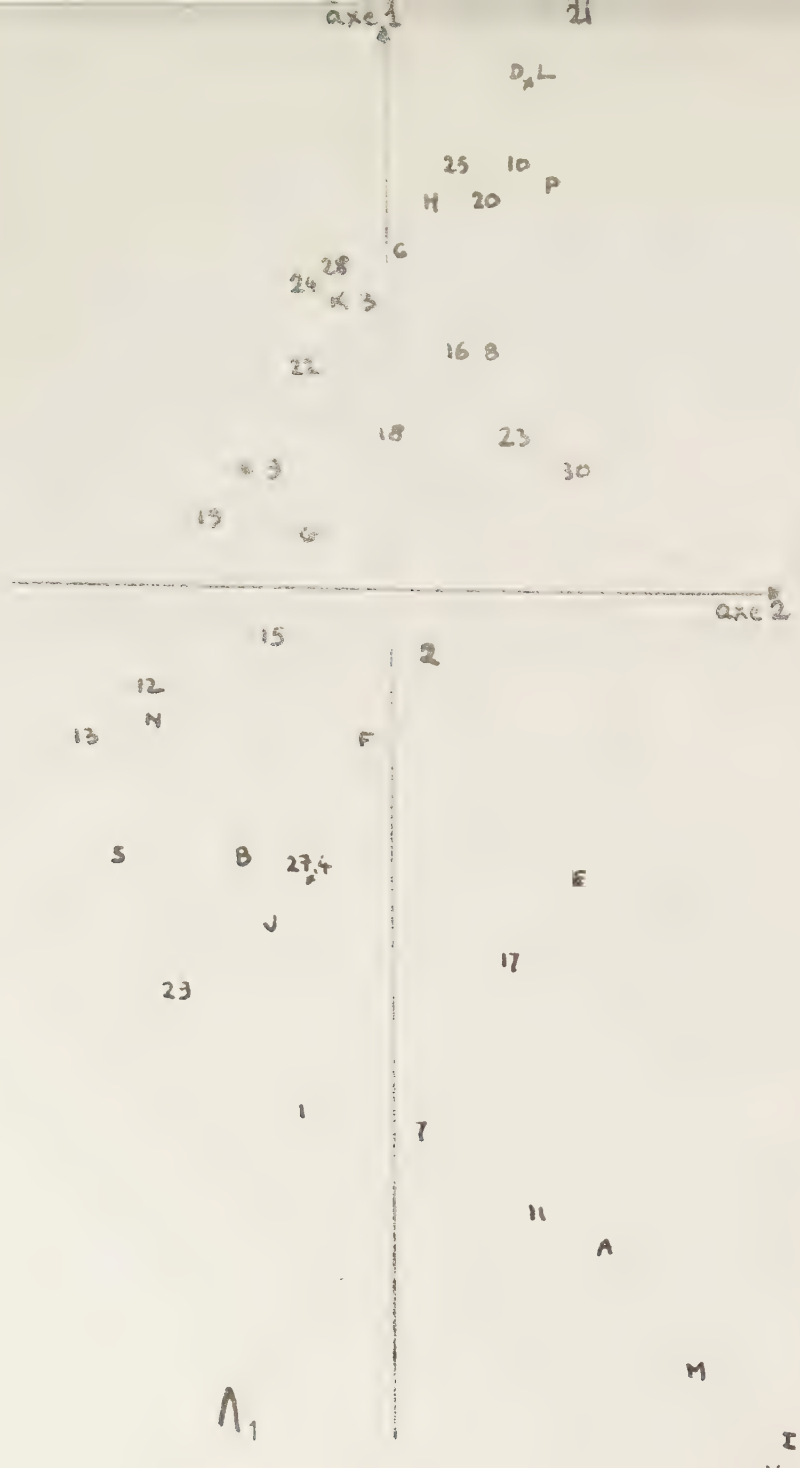
CI 23

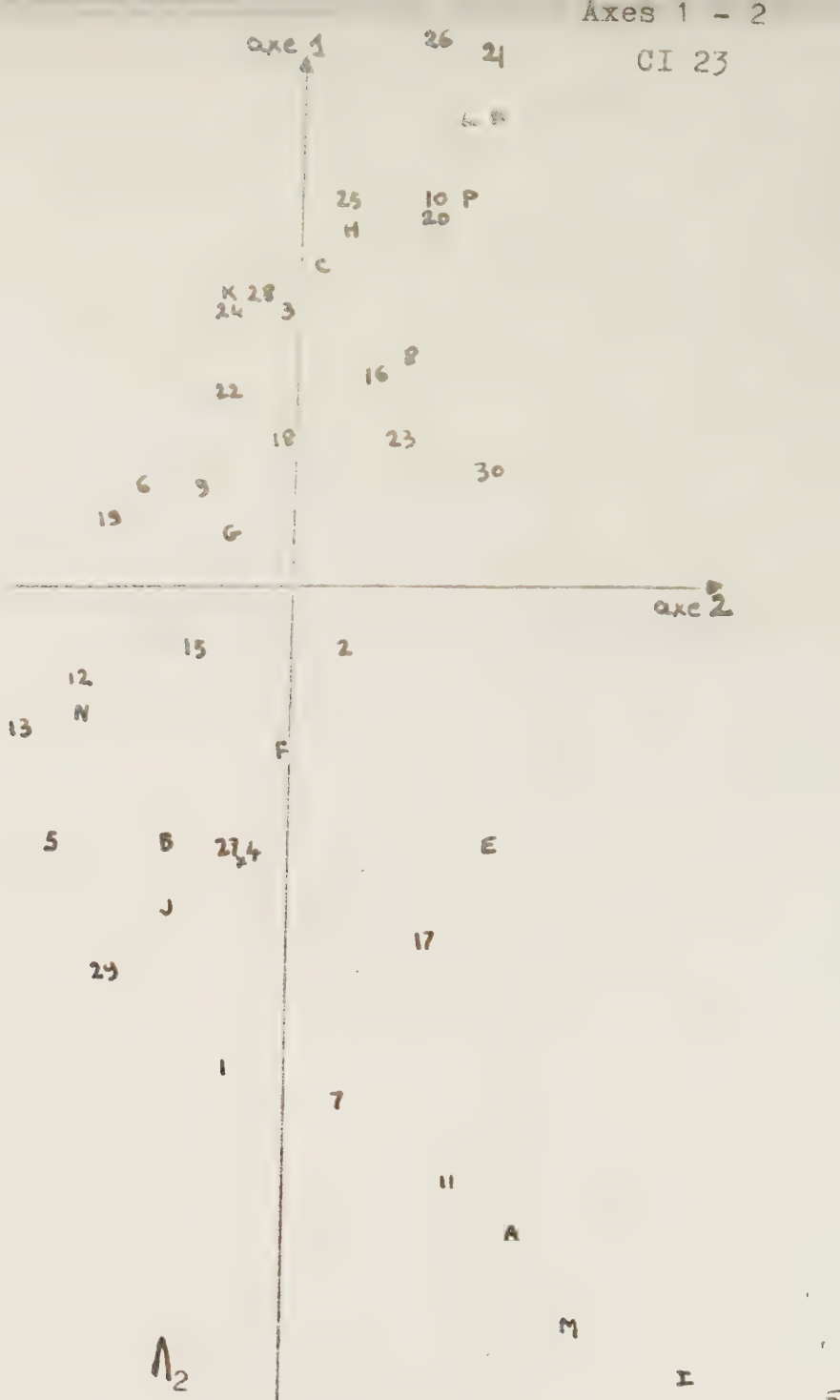


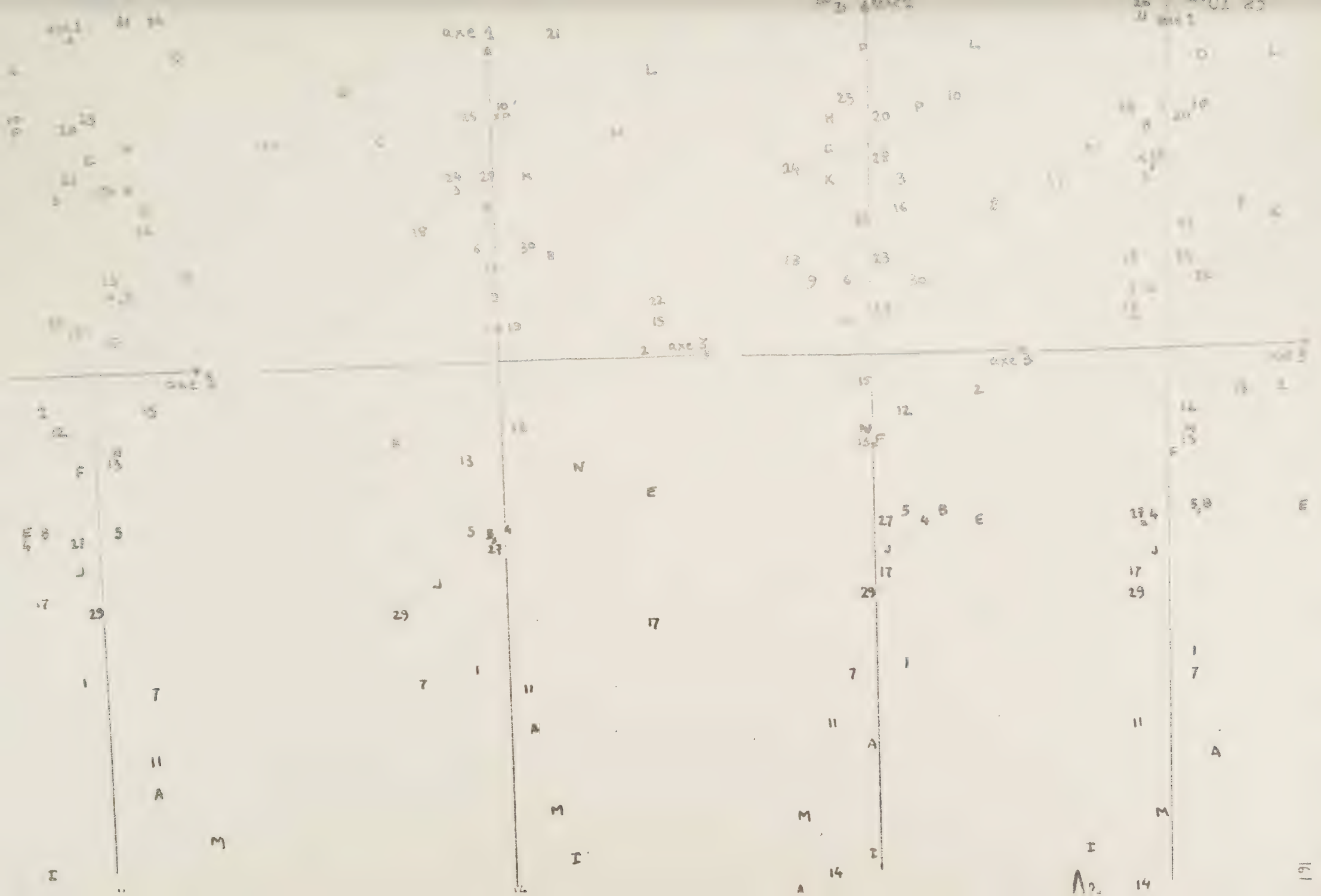


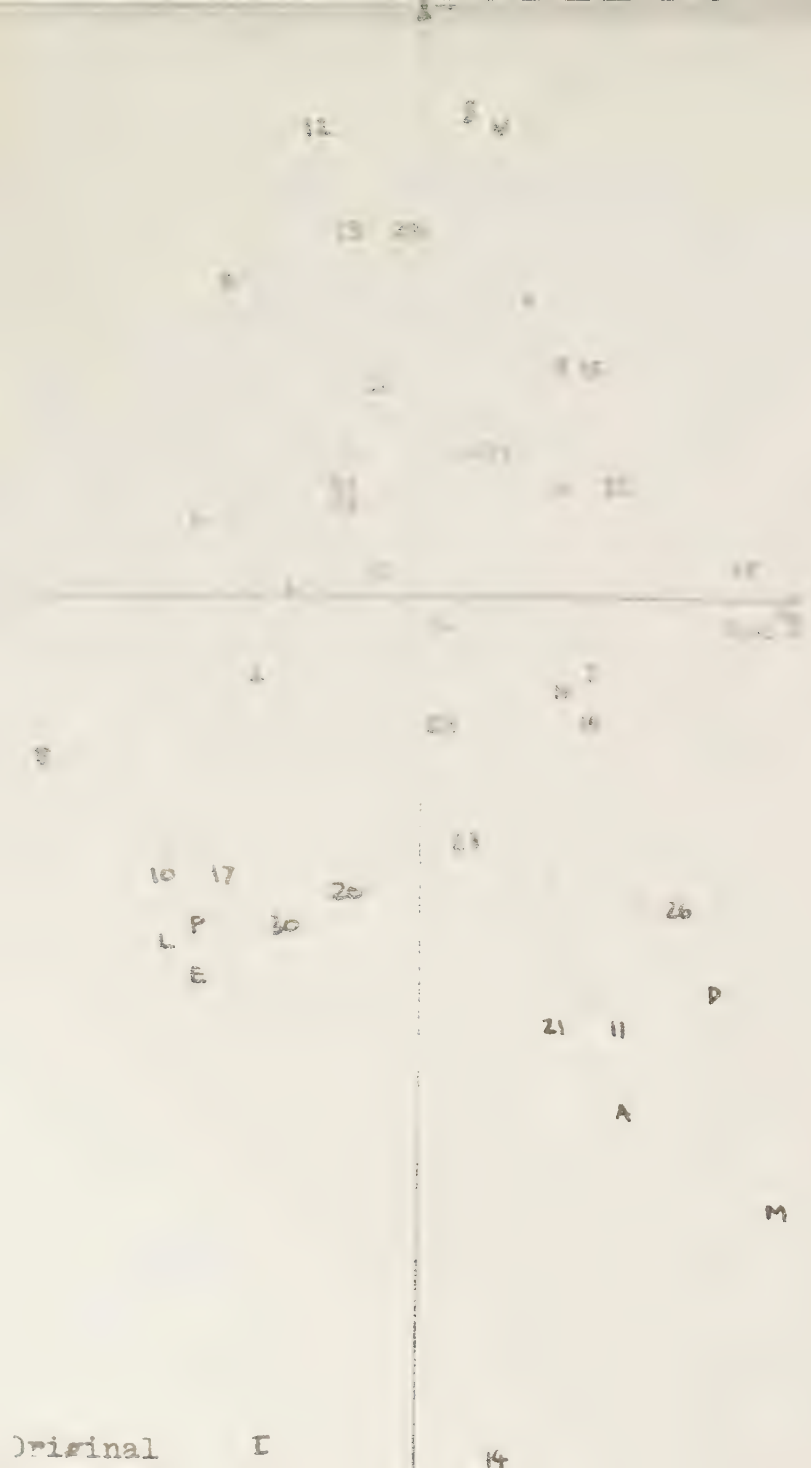
Axes 1 - 2

CI 23

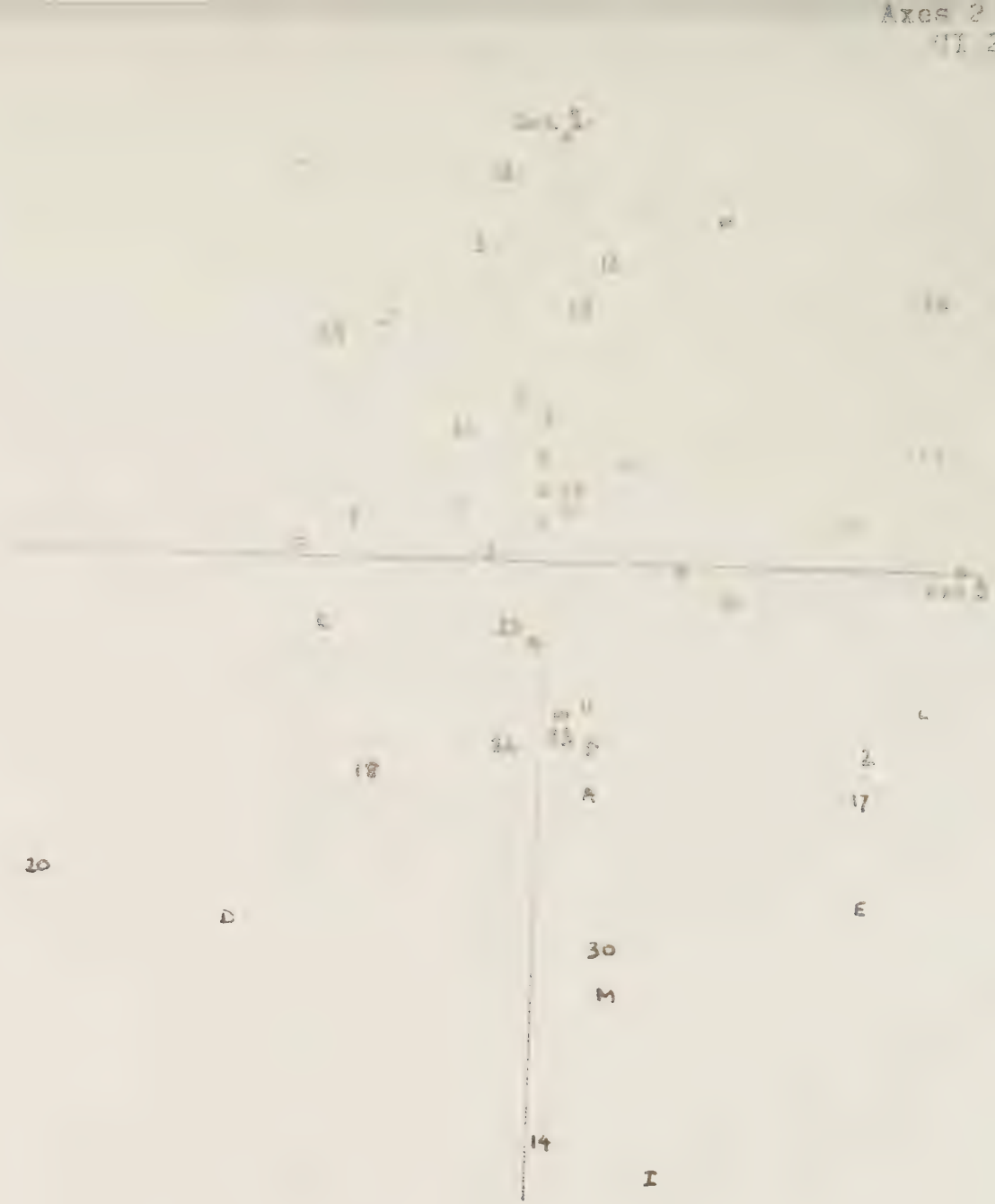

axe 1
axe 2
D L
25 10 P
H 20
G
24 28
K 3
22
16 8
18 23
3 30
19
G
15
2
12
N
13
F
5
B
27,4
E
J
17
23
1
7
11
A
M
I
Λ_1


axe 1
axe 2
26 21
25 10 P
H 20
C
K 28
24 3
16 8
22
18 23
30
6 9
19
G
15
2
12
N
13
F
5
B
27,4
E
J
17
29
1
7
11
A
M
I
Λ_2

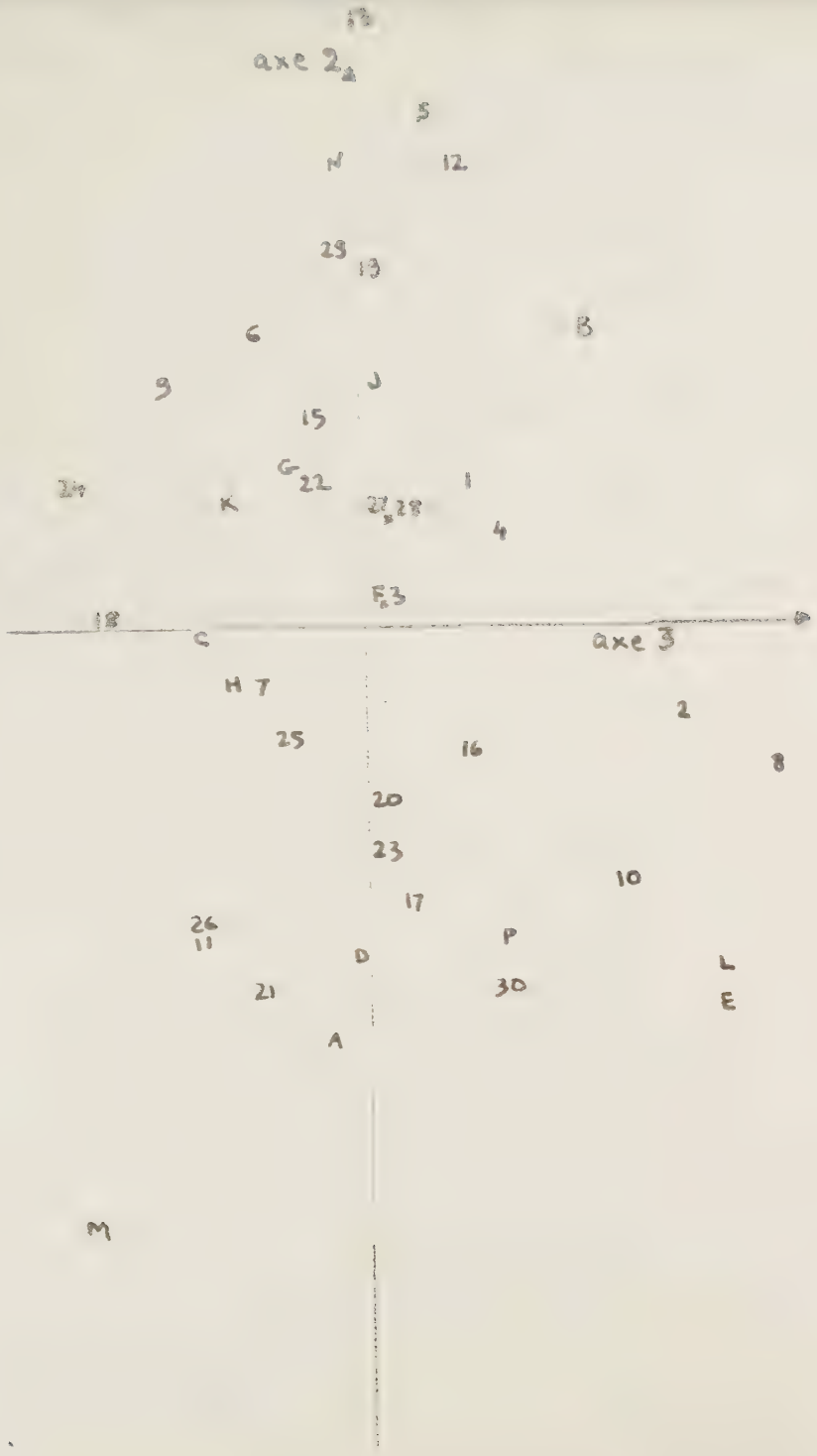

axe 1
axe 2
axe 3

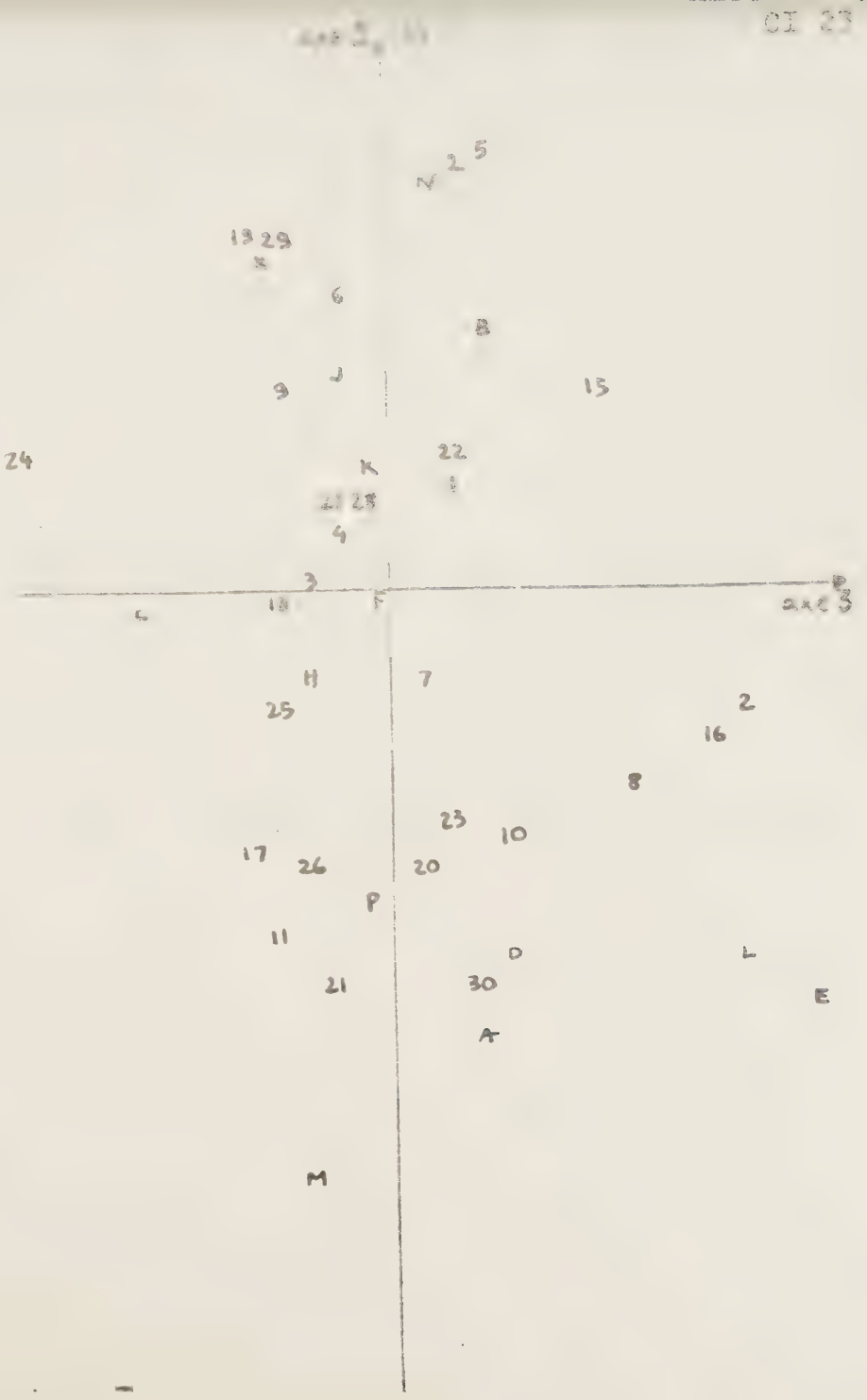


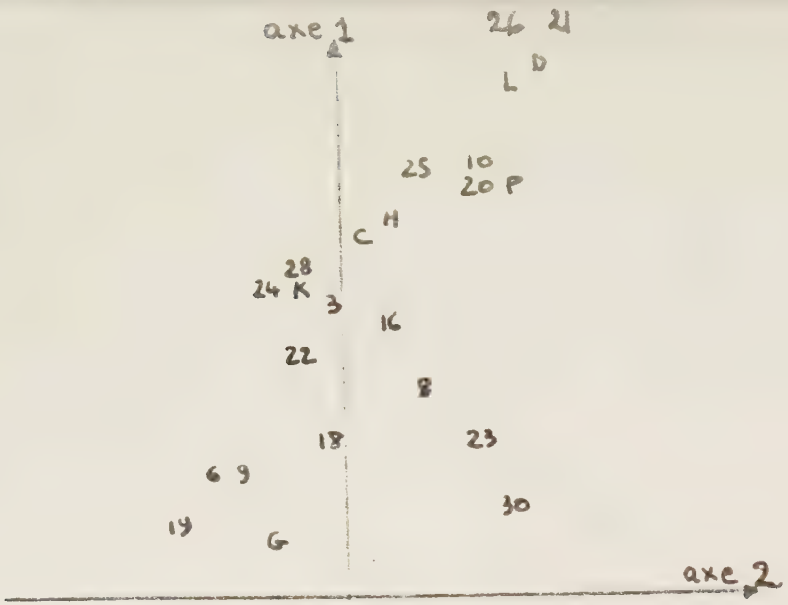
Original I

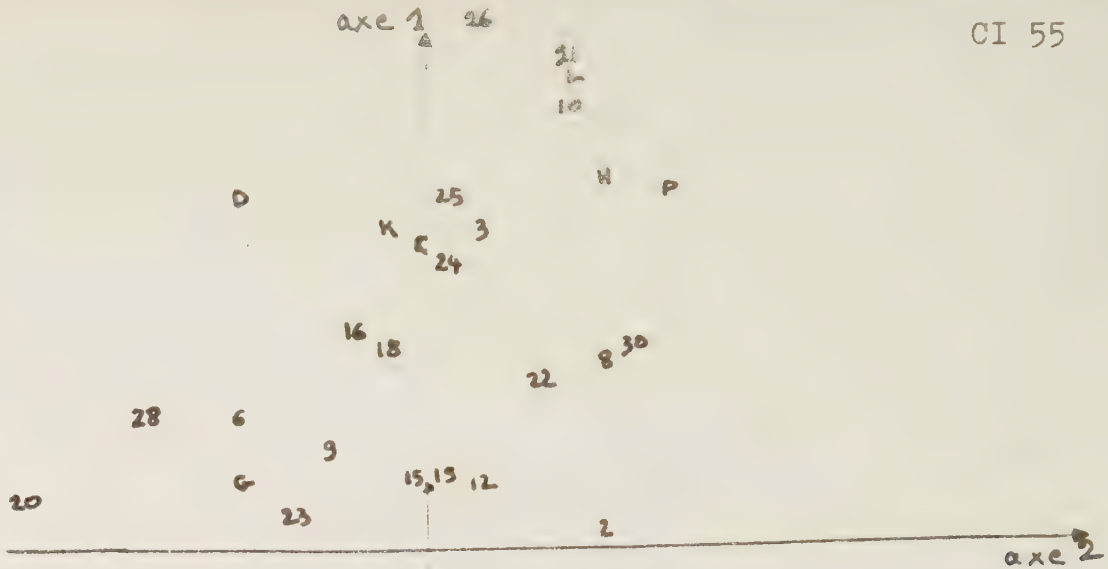


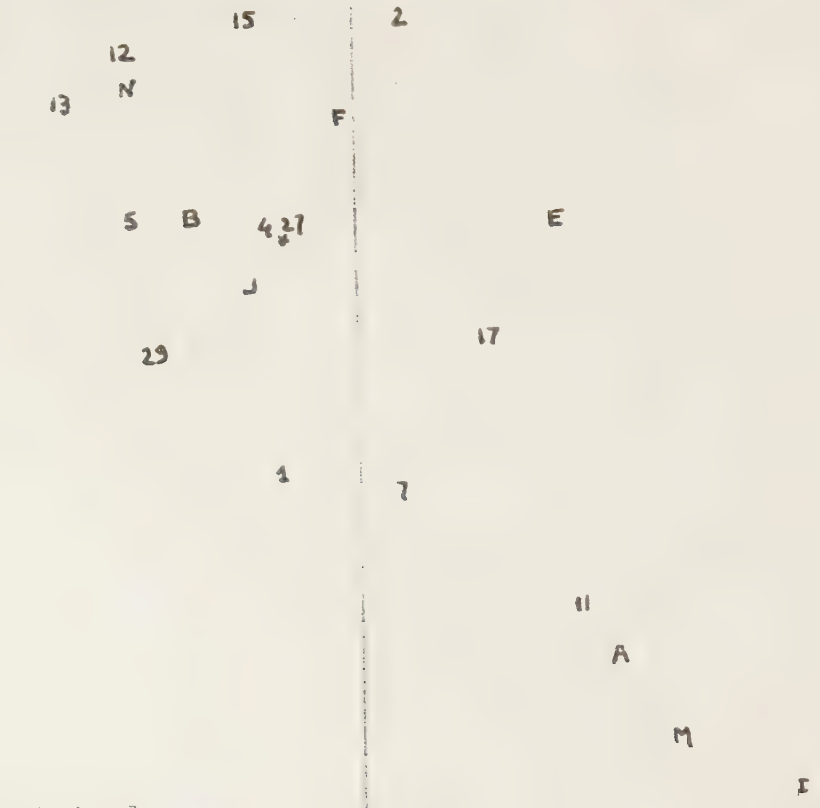
Trous

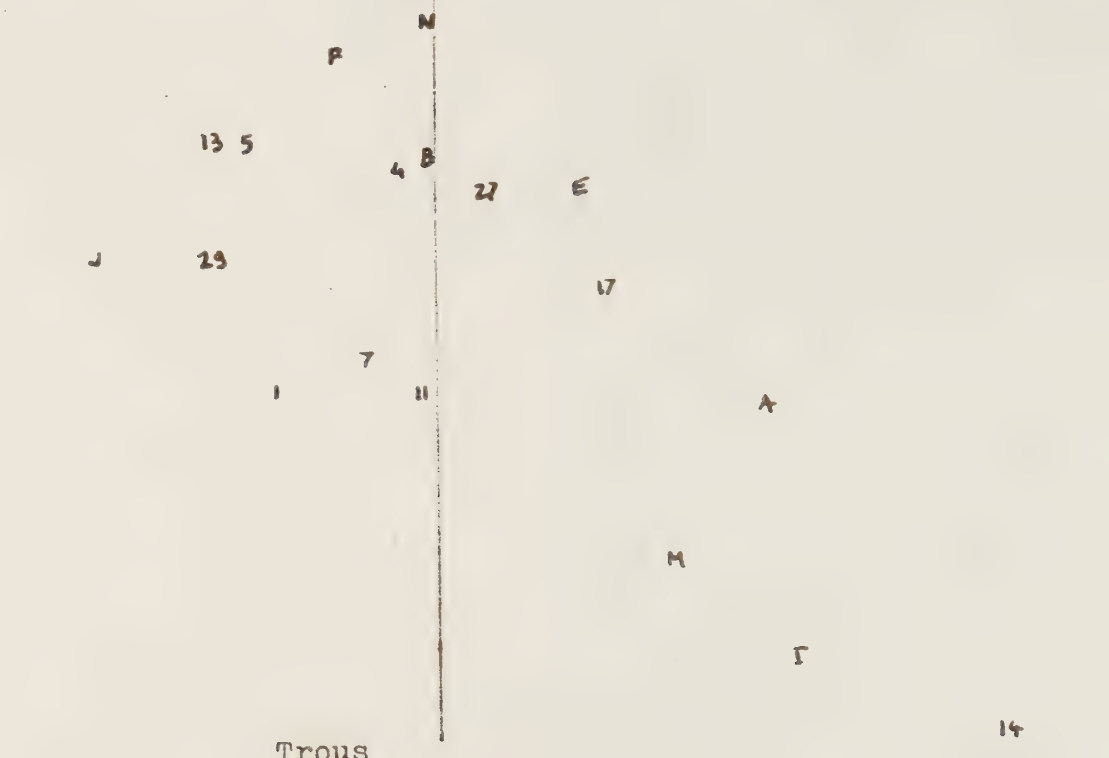

axe 2
axe 3
5
N
12
29
13
6
B
9
J
15
G
22
24
K
1
27
28
4
F
3
18
C
H
7
25
16
2
8
20
23
10
17
26
11
P
D
L
21
30
E
A
M


axe 3
5
N
2
13
29
6
B
9
J
15
24
22
K
1
27
28
4
3
18
C
F
H
7
25
2
16
8
23
10
17
26
20
P
11
D
L
21
30
E
A
M

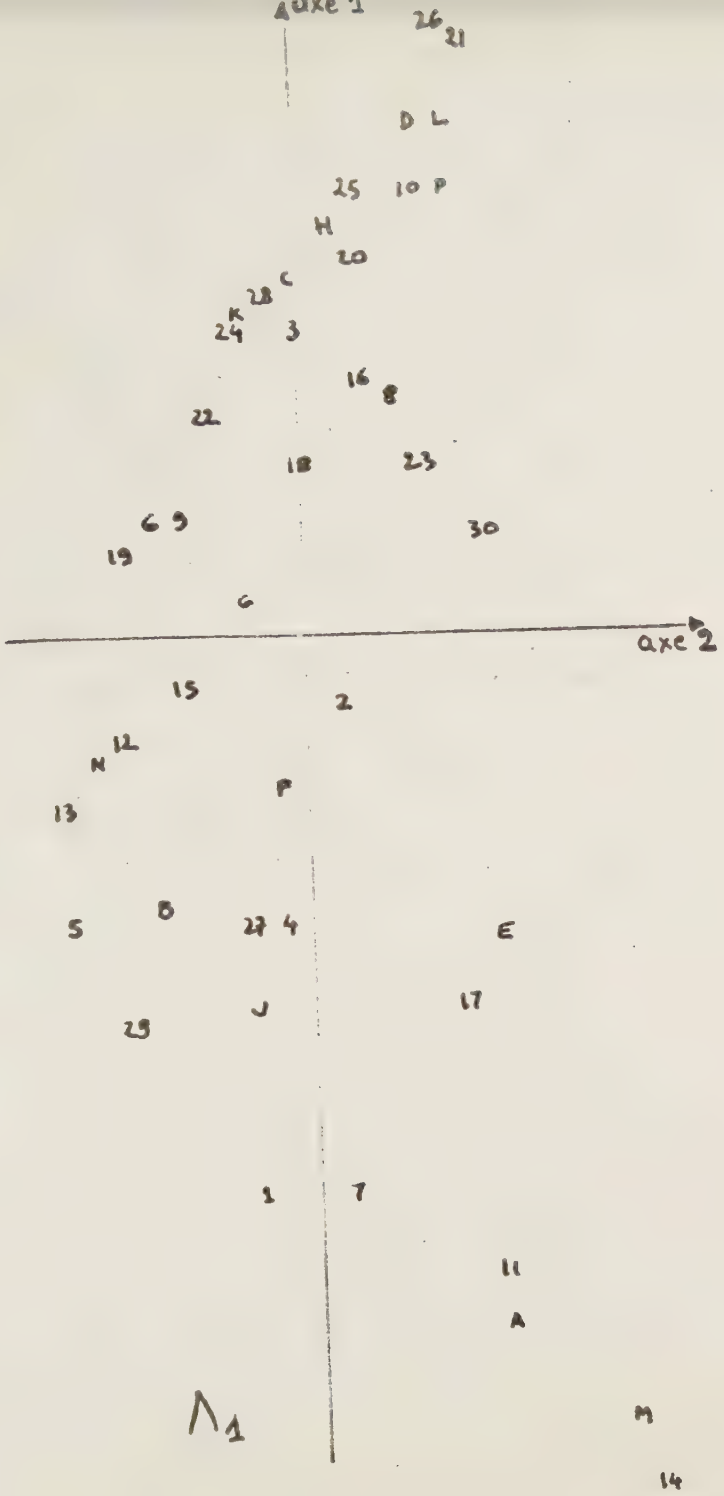

axe 1
axe 2
26 21
L D
25 10
20 P
C H
28
24 K
3
16
22
B
18
23
6 9
30
19
G

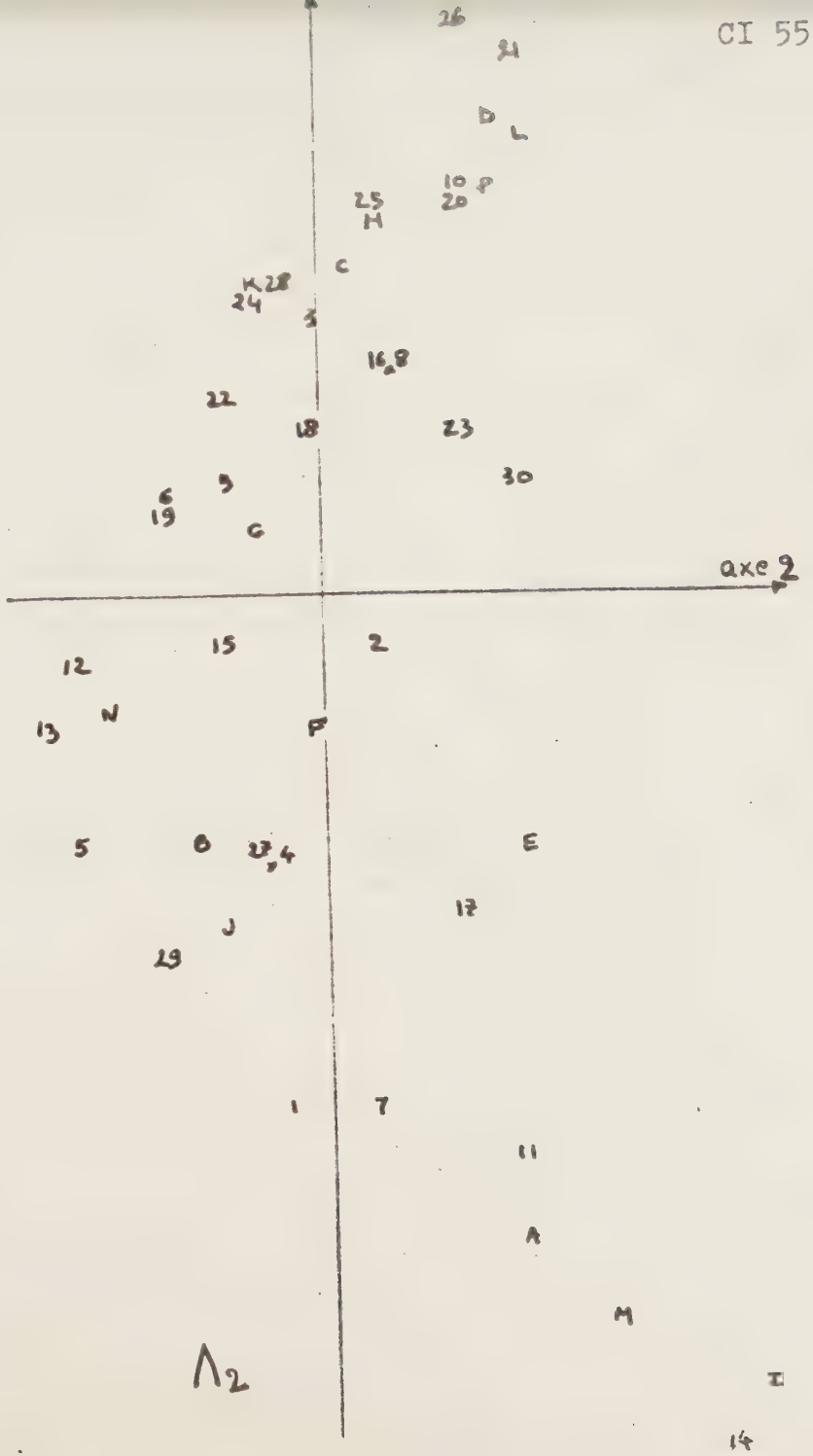

axe 1
axe 2
26
21
L
10
H
P
D
25
K
3
E
24
16
18
8
30
22
28
6
9
G
15
15
12
20
23
2

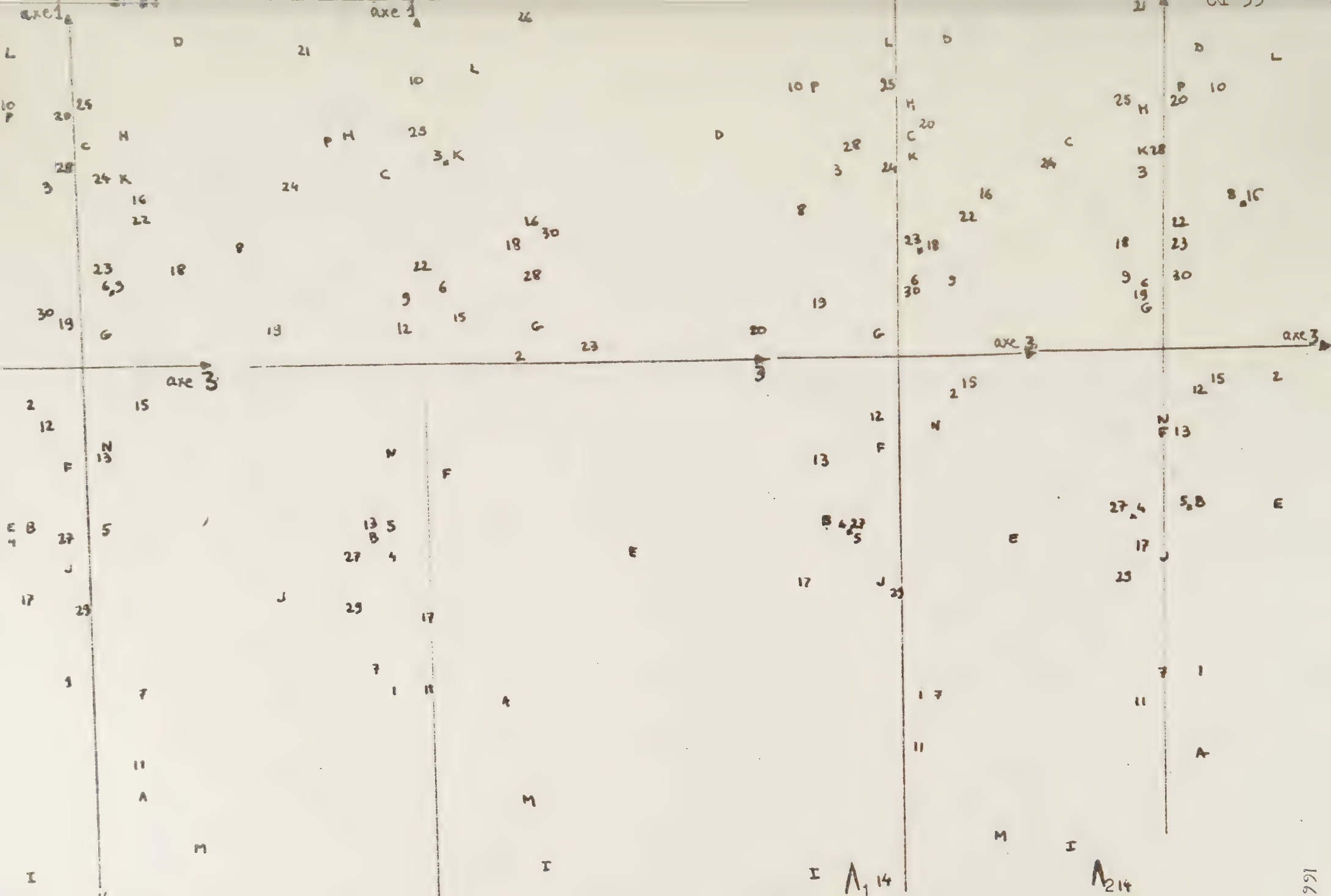

15
2
12
N
13
F
5
B
4
27
E
J
17
29
1
7
11
A
M
I


N
P
13
5
4
B
27
E
J
29
17
7
1
11
A
M
I
14

Trous


axe 1
axe 2
26
21
D L
25 10 P
H
20
C
28
K
24
3
16
8
22
18
23
6
9
19
30
G
15
2
12
N
P
13
5
B
27
4
E
J
17
29
1
7
11
A
M
I
14
Λ_1


axe 2
26
21
D
L
10
P
25
20
H
C
K
28
24
3
16
8
22
18
23
30
6
19
G
15
2
12
N
P
13
5
B
27
4
E
17
J
29
1
7
11
A
M
I
14
Λ_2


axe 1
axe 3
axe 1
axe 3
axe 3
axe 3

Axes 2 - 3

CI 55

Trous

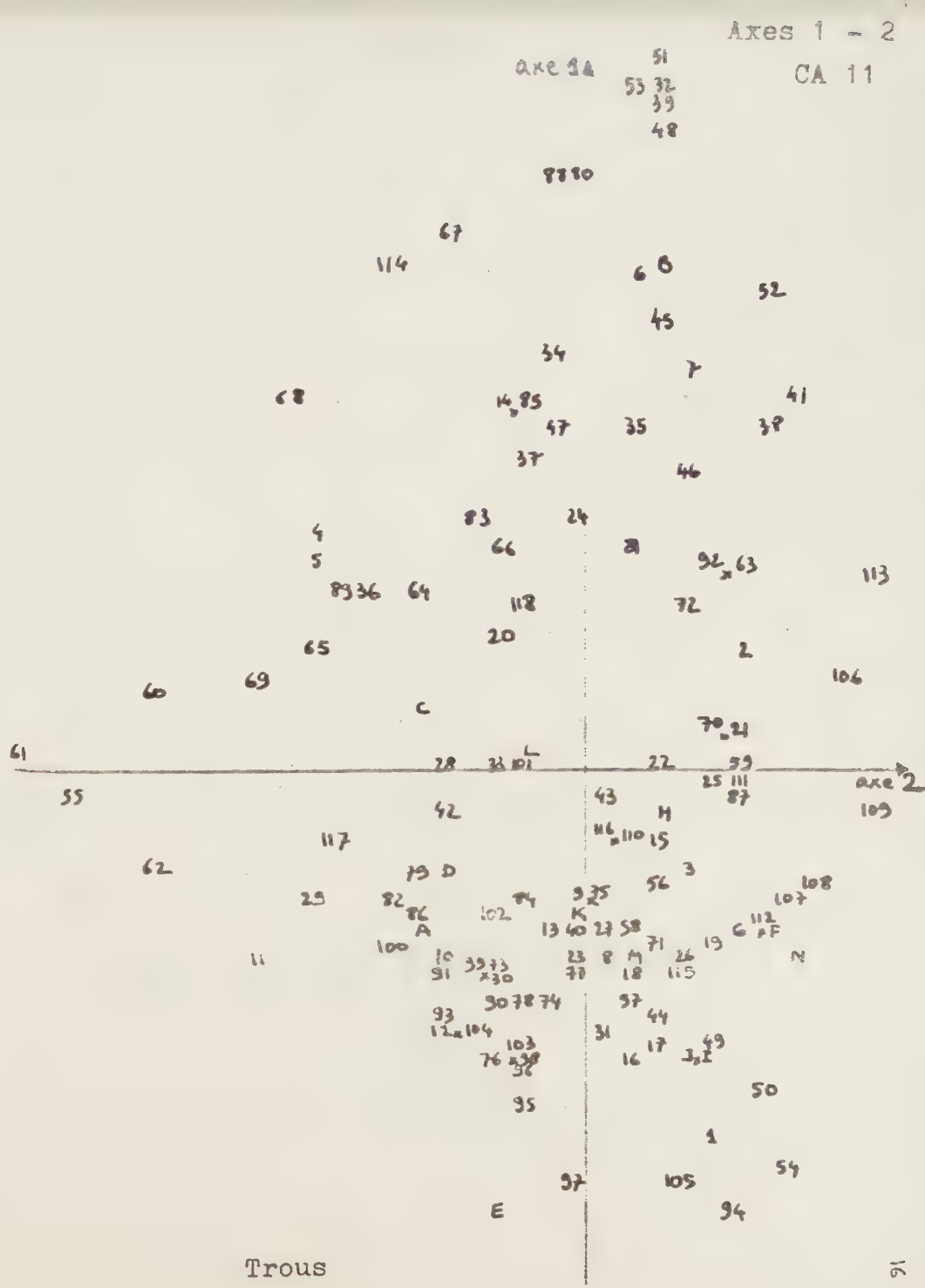
Axes 2 - 3

CI 55

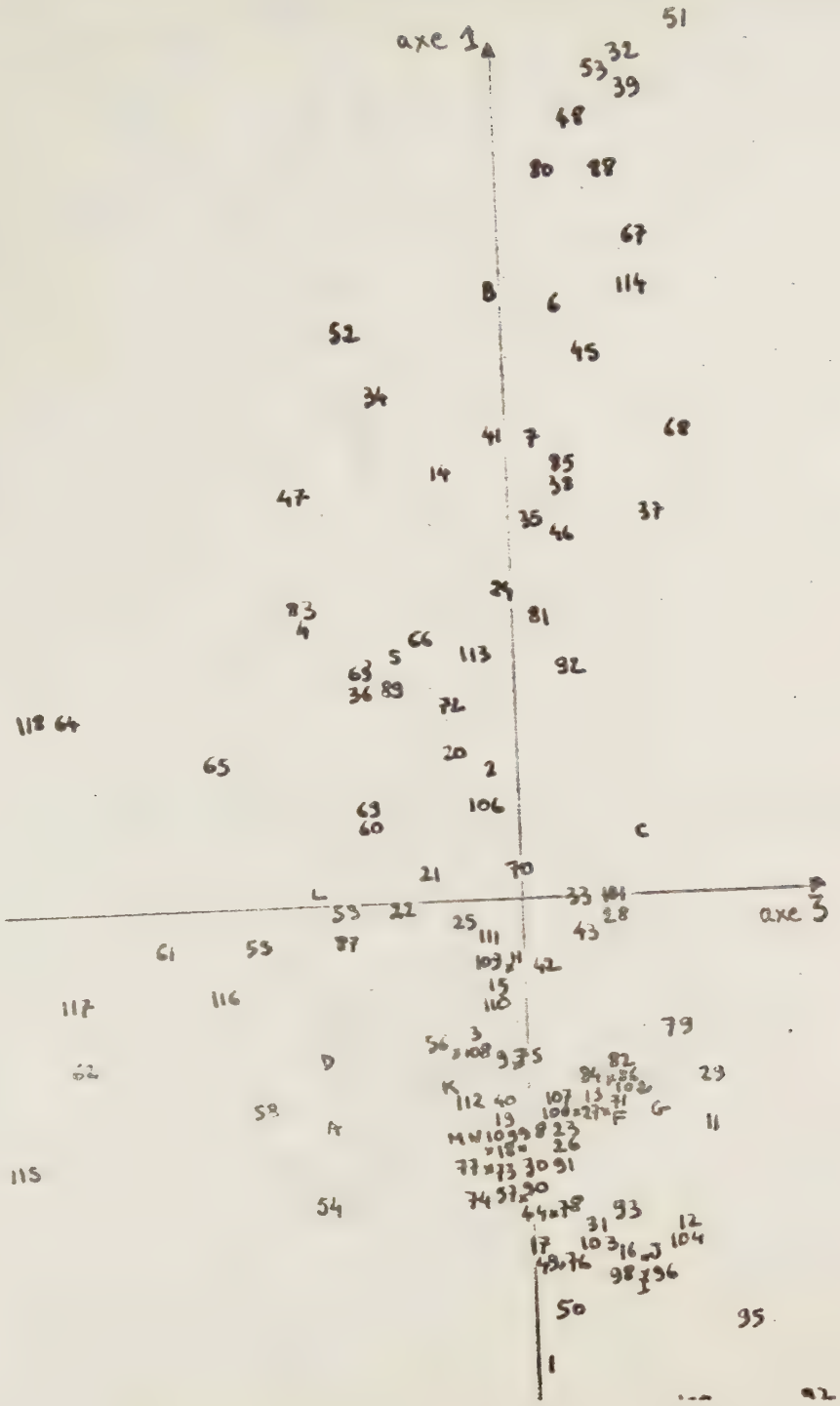
Λ_1

Λ_2

axe 1
axe 2
Original

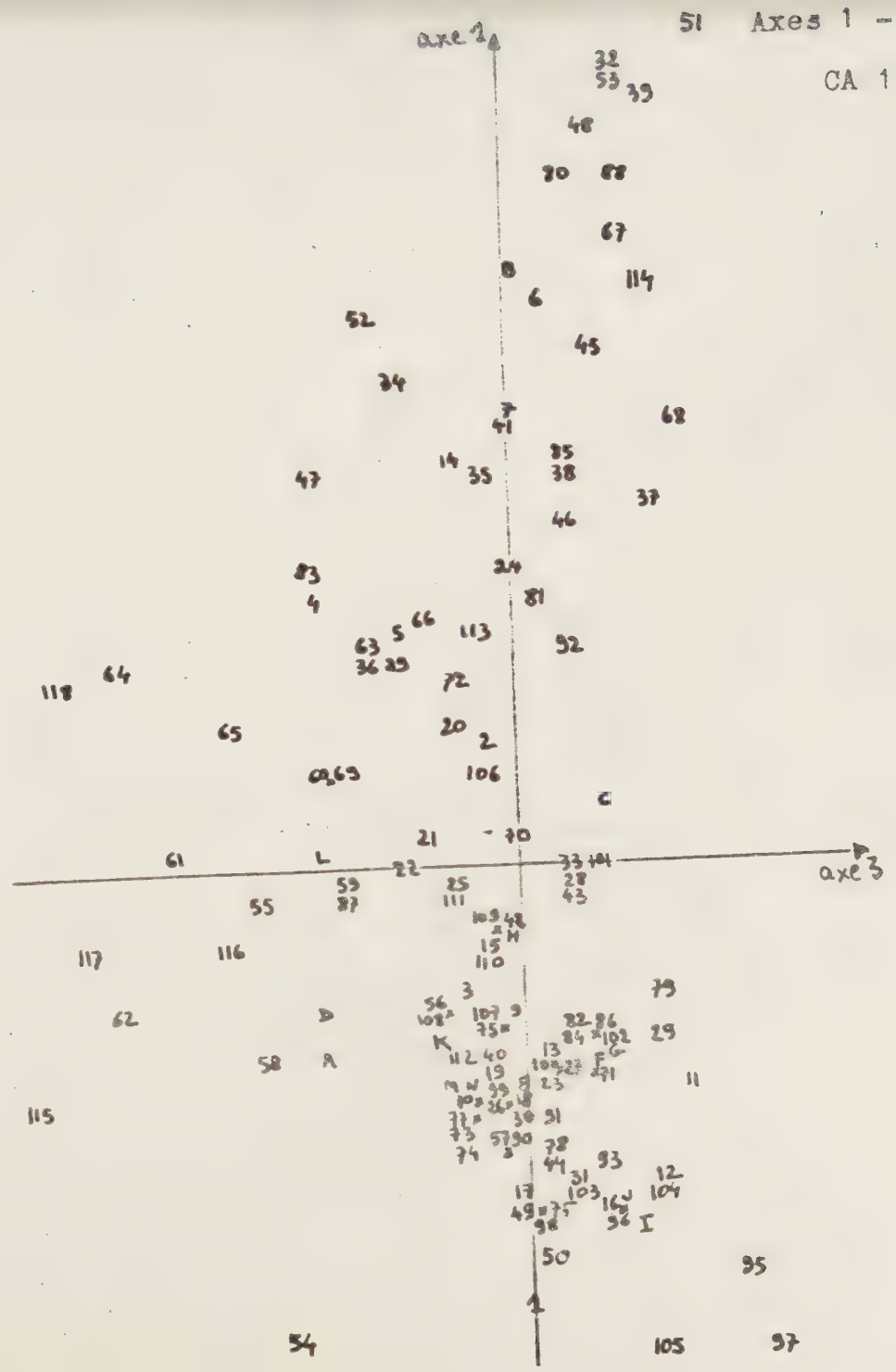

Axes 1 - 2
CA 11
axe 1
axe 2
Trous

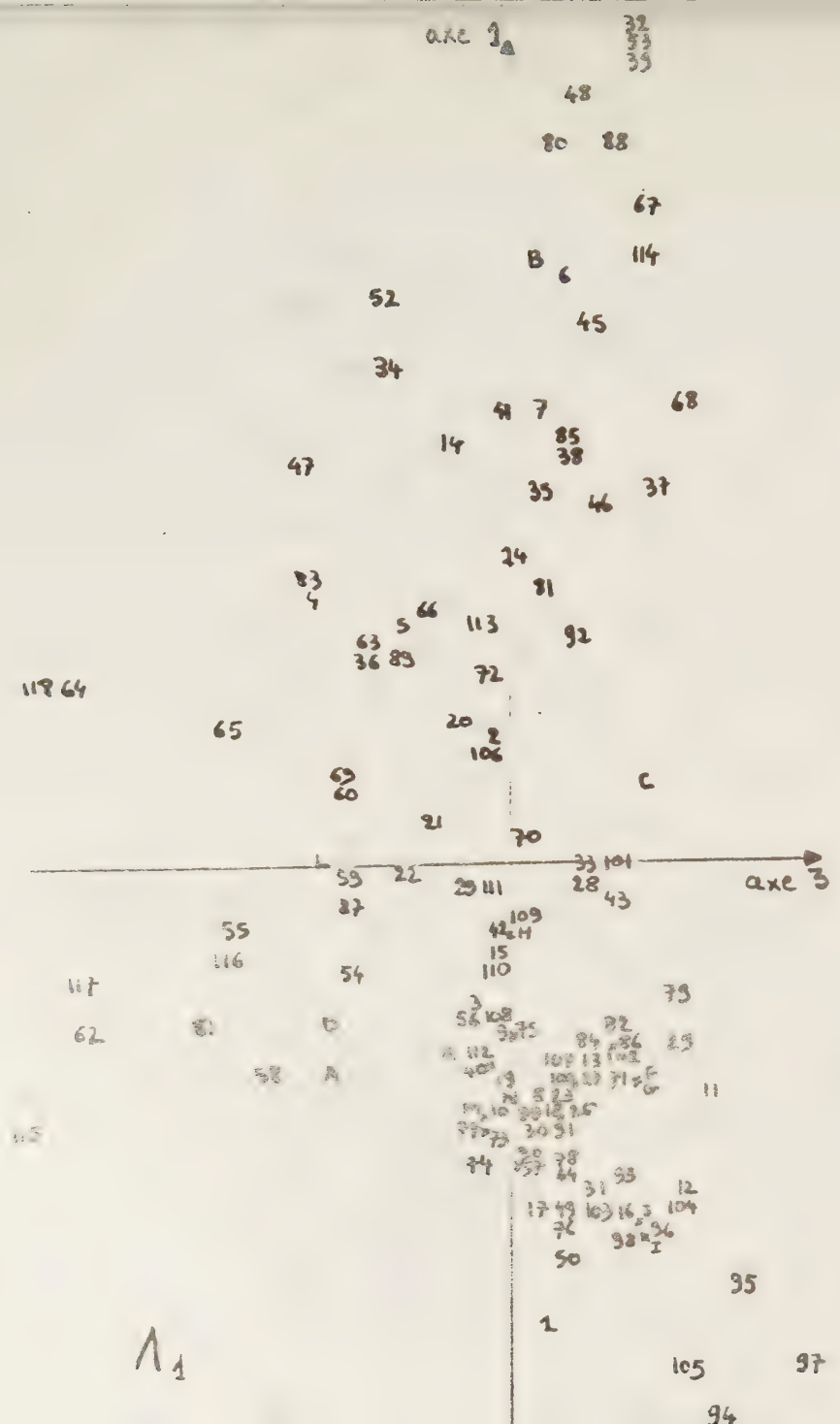
axe 1
axe 2
Axes 1 - 2
CA 11

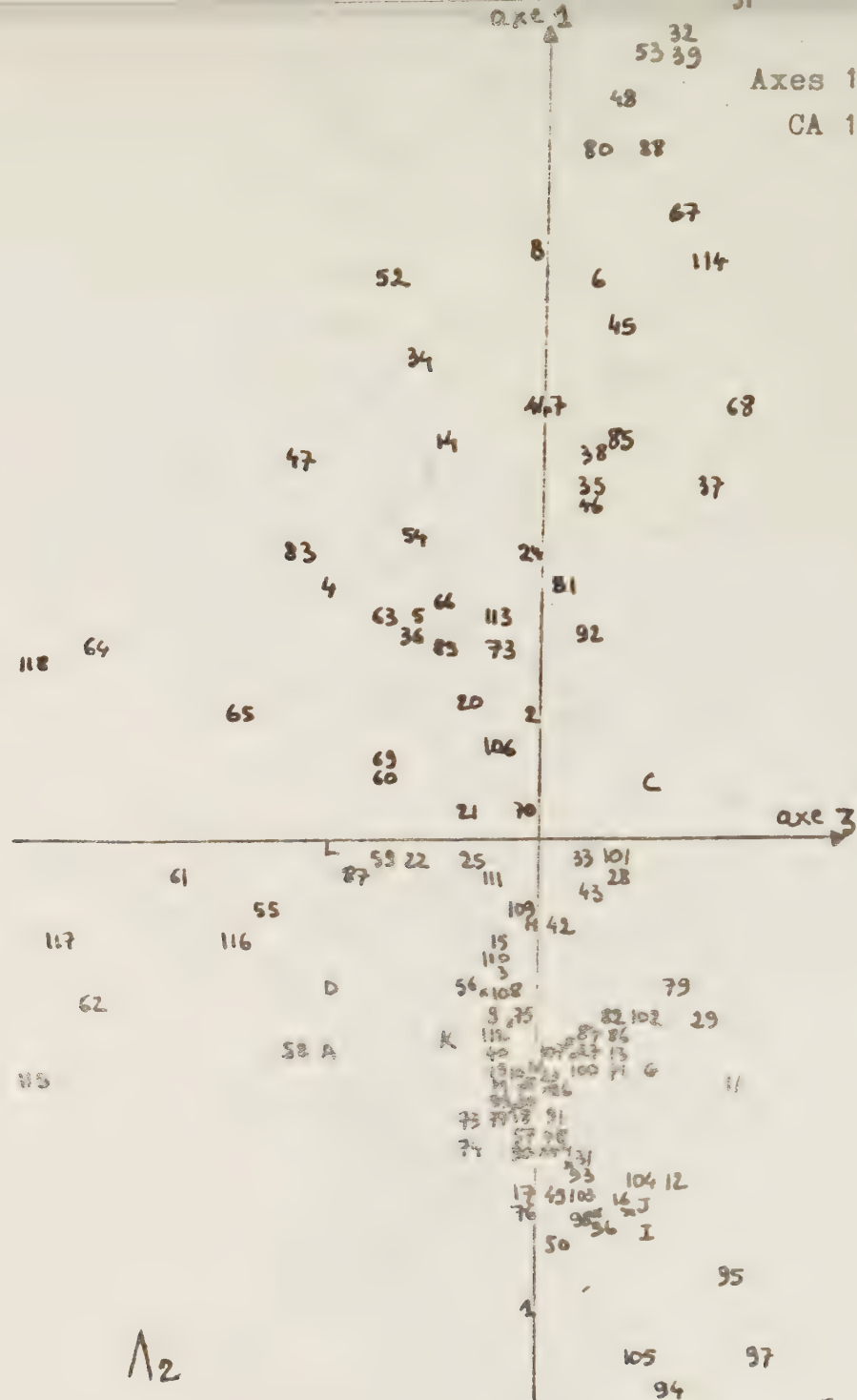


51 Axes 1 - 3

CA 11

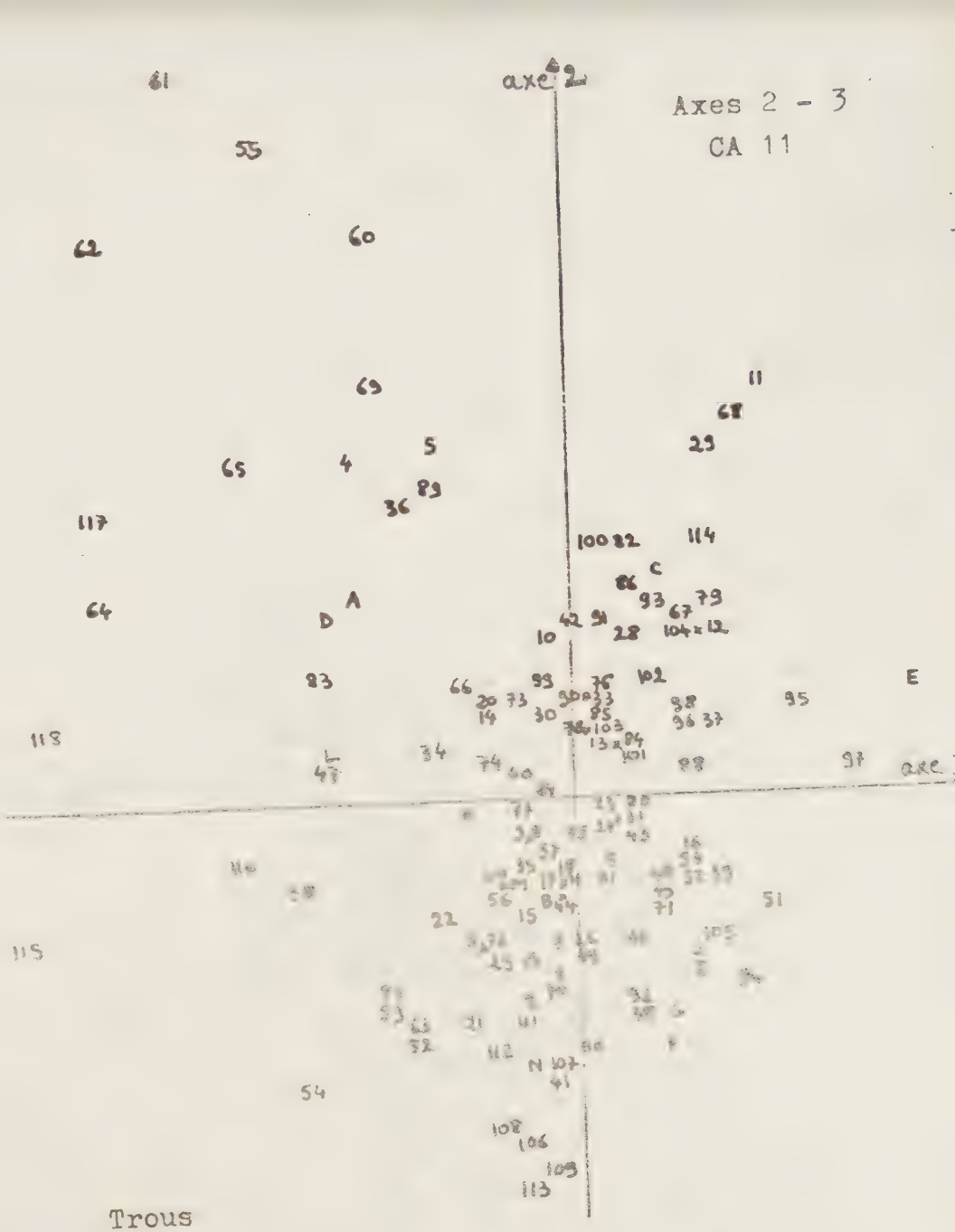






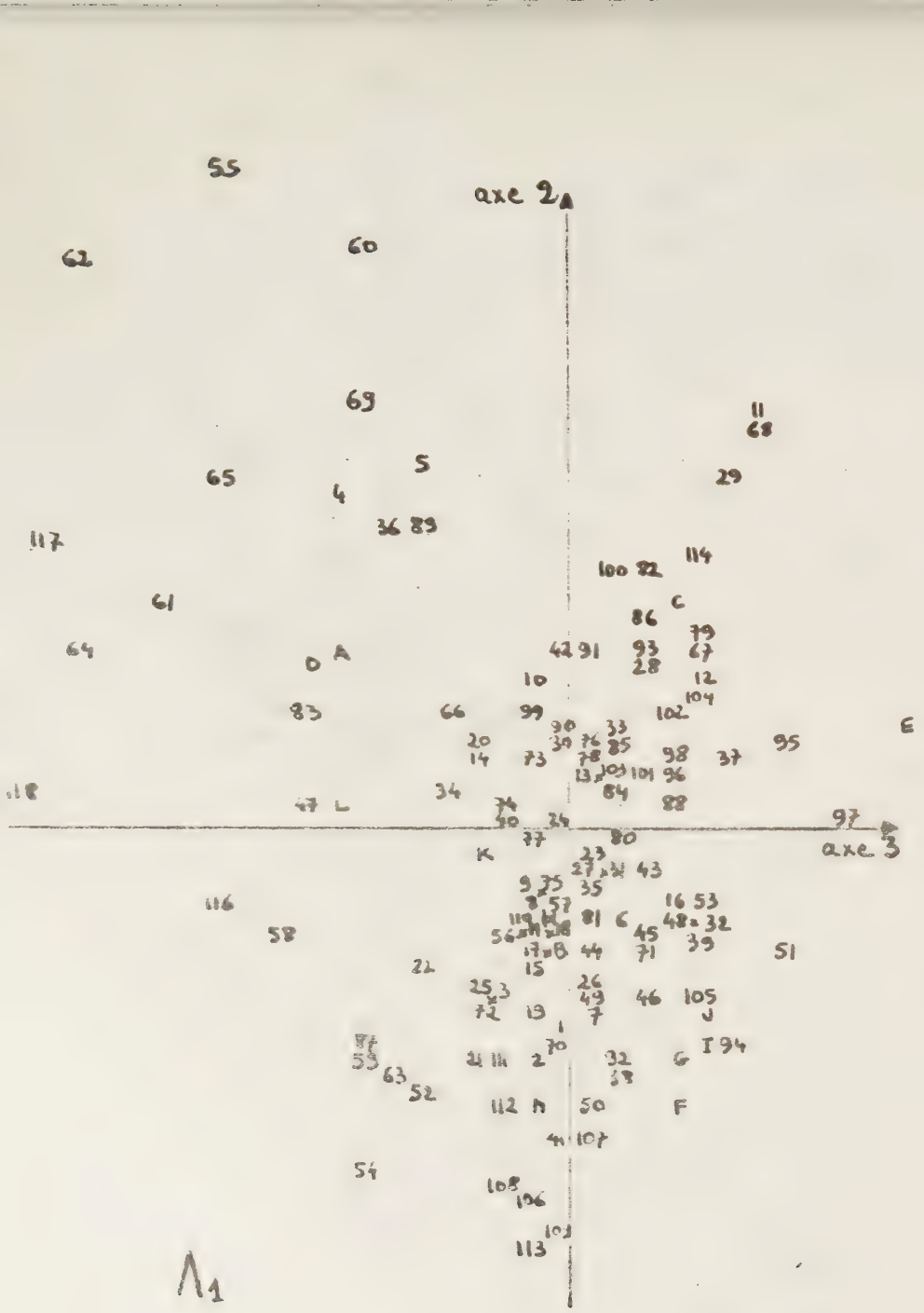
Axes 1-3
CA 11

Original



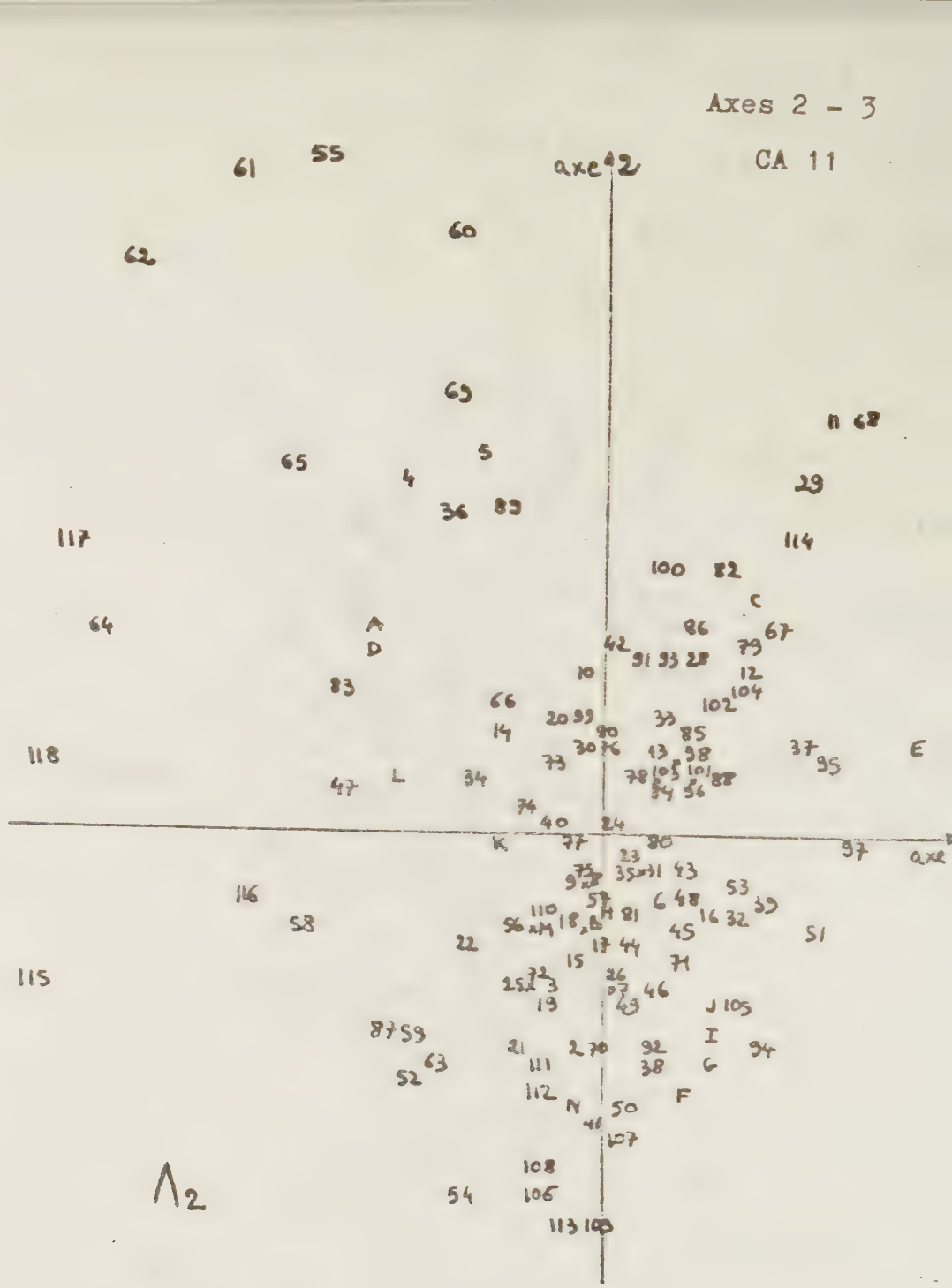
Axes 2 - 3
CA 11

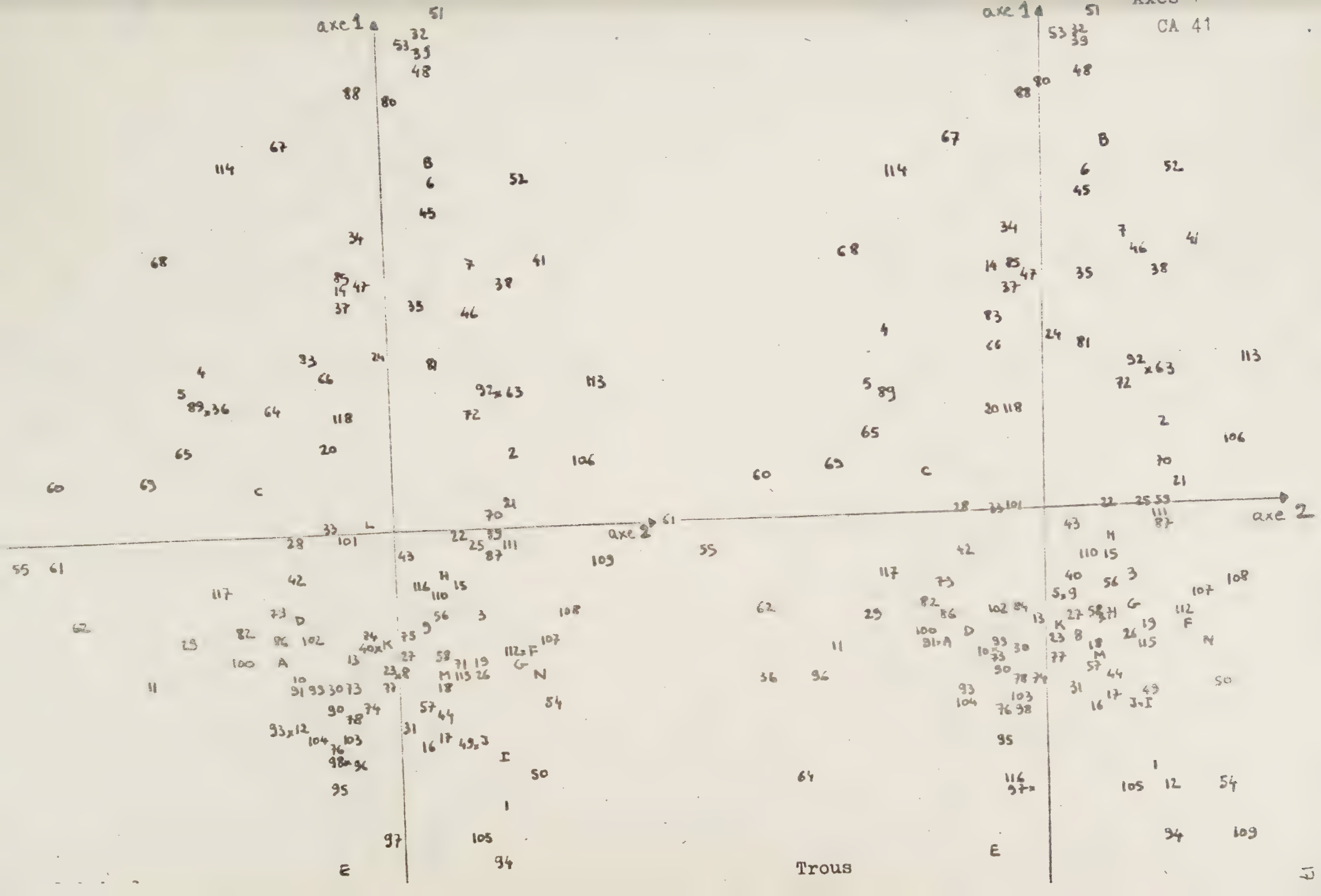
Trous

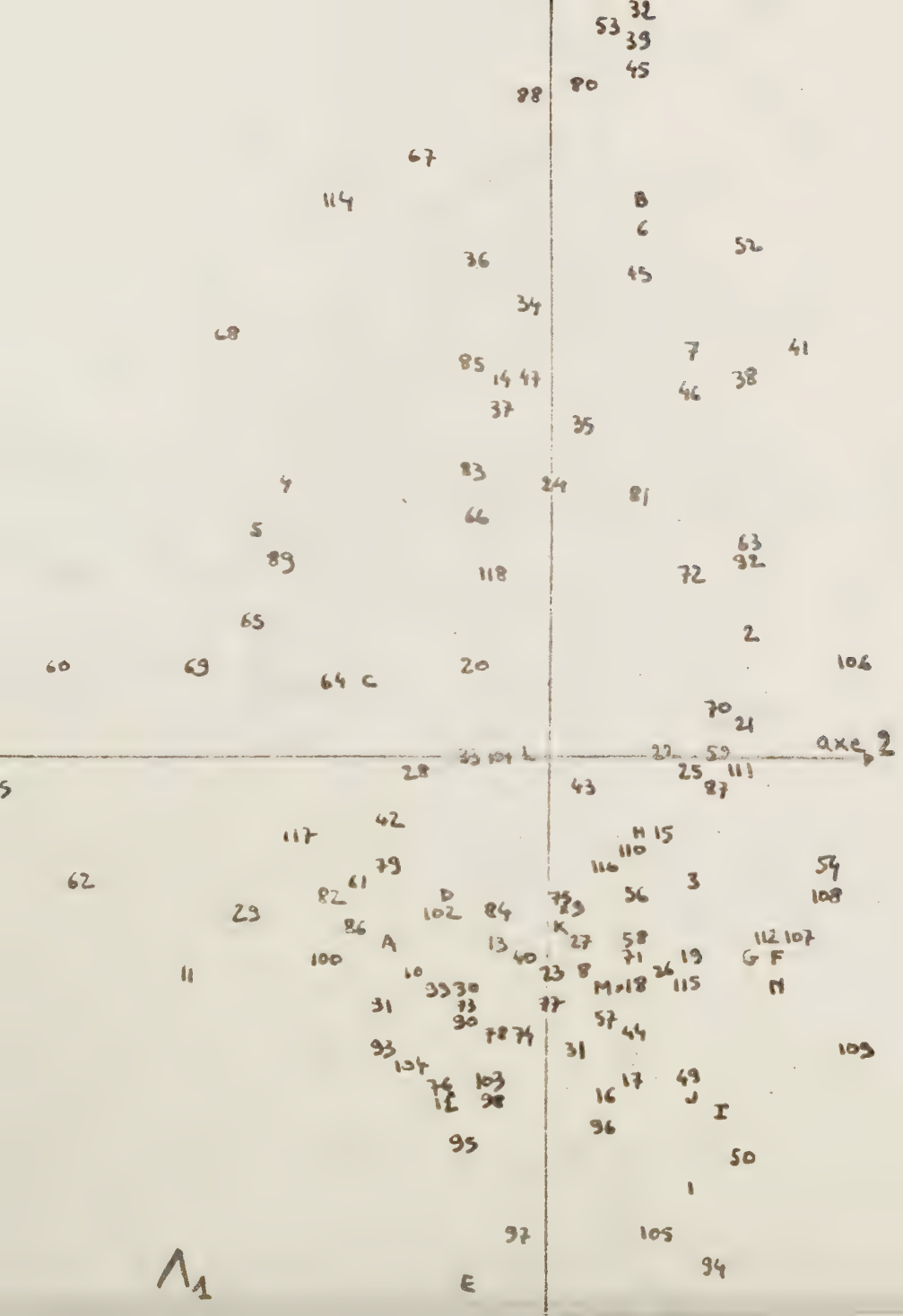


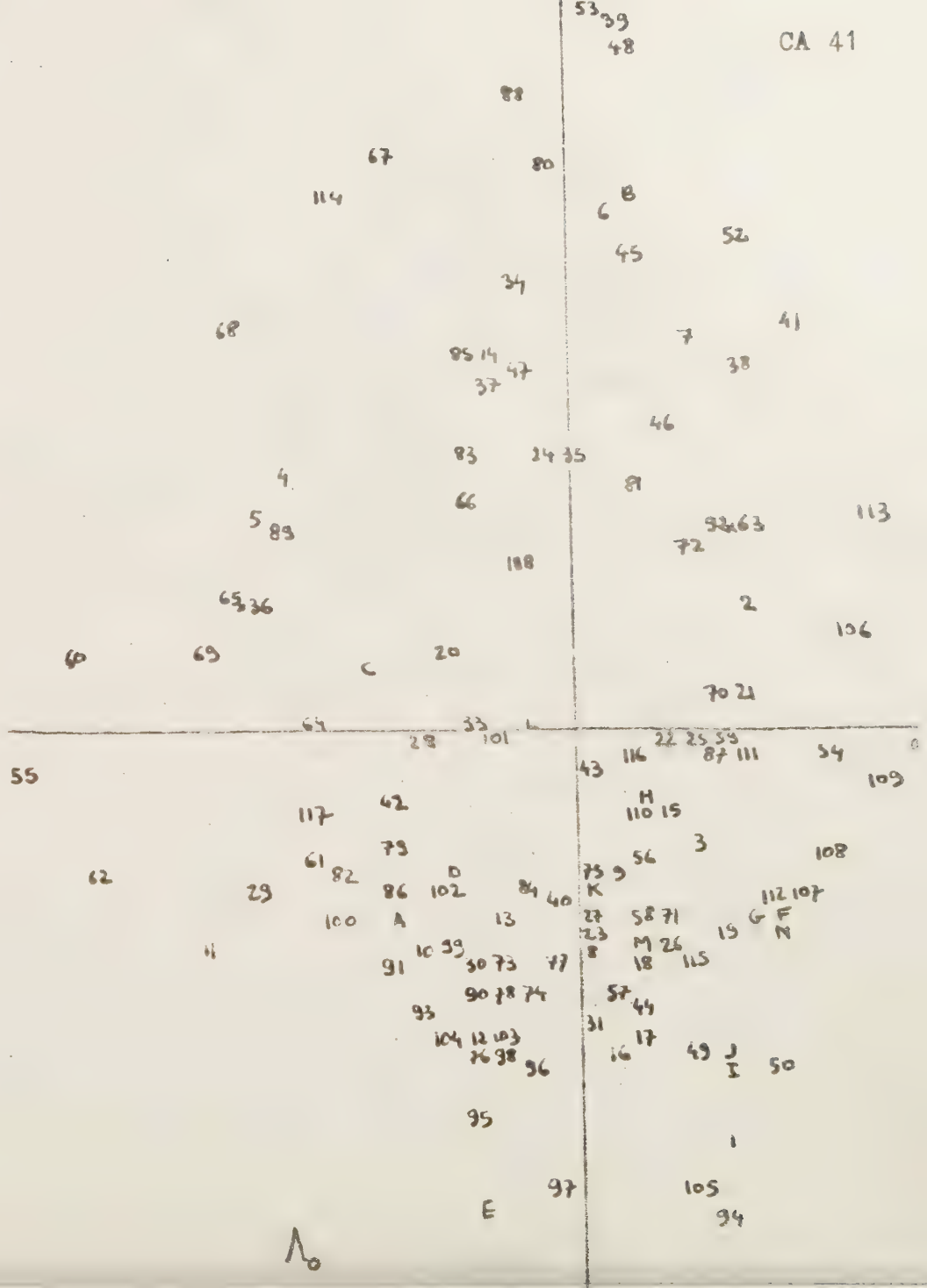
Axes 2 - 3

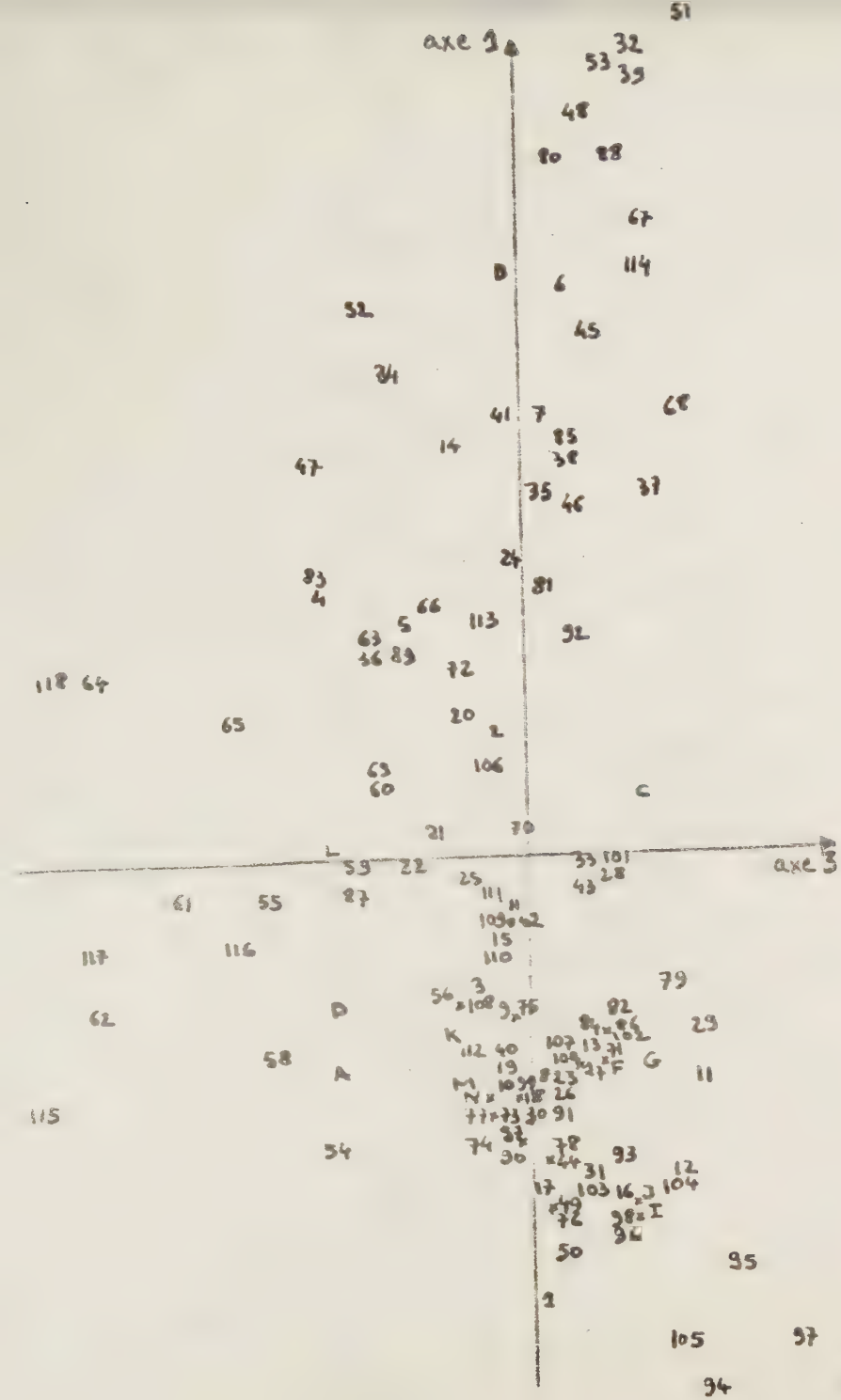
CA 11

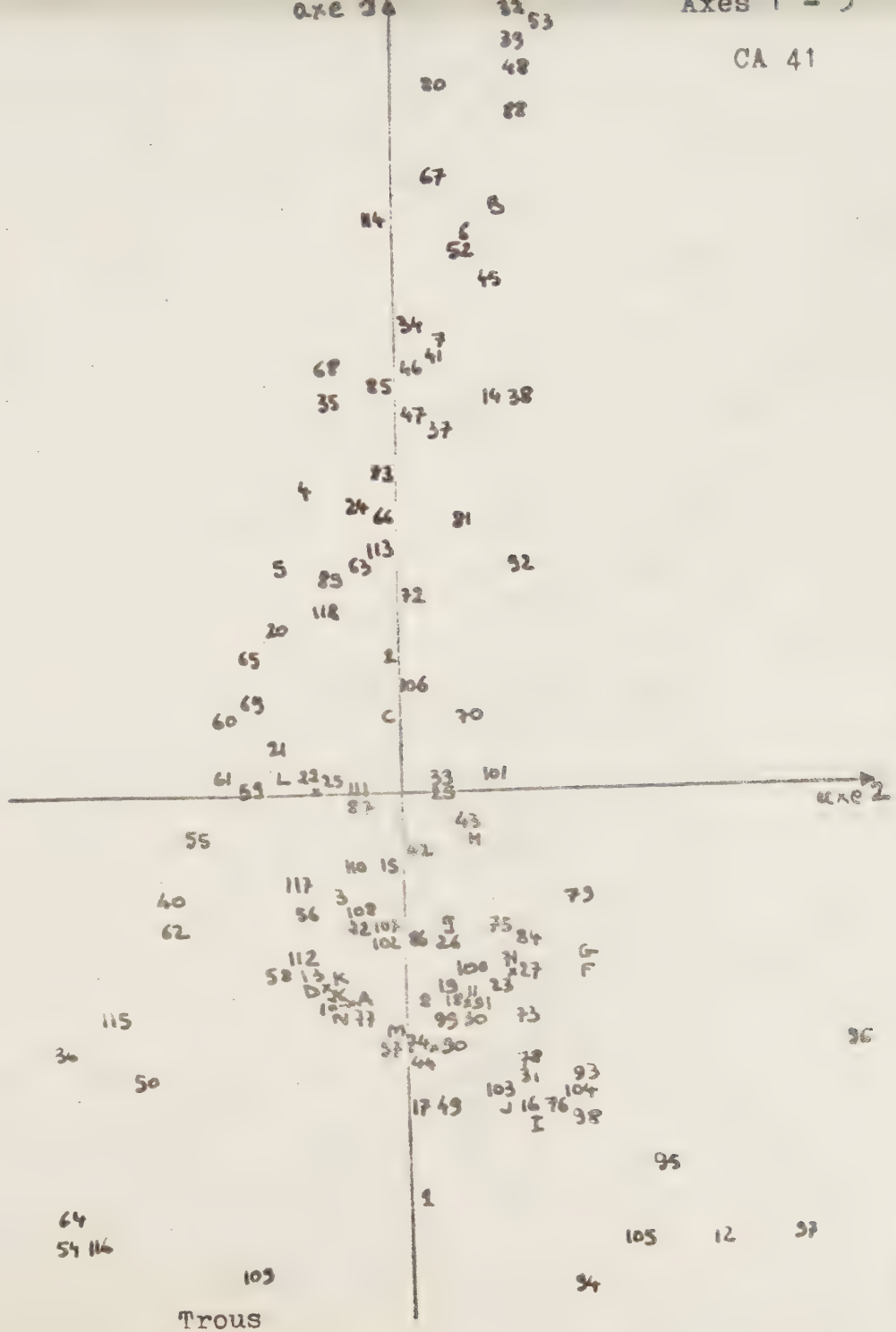



axe 1
axe 2
CA 41
Trous

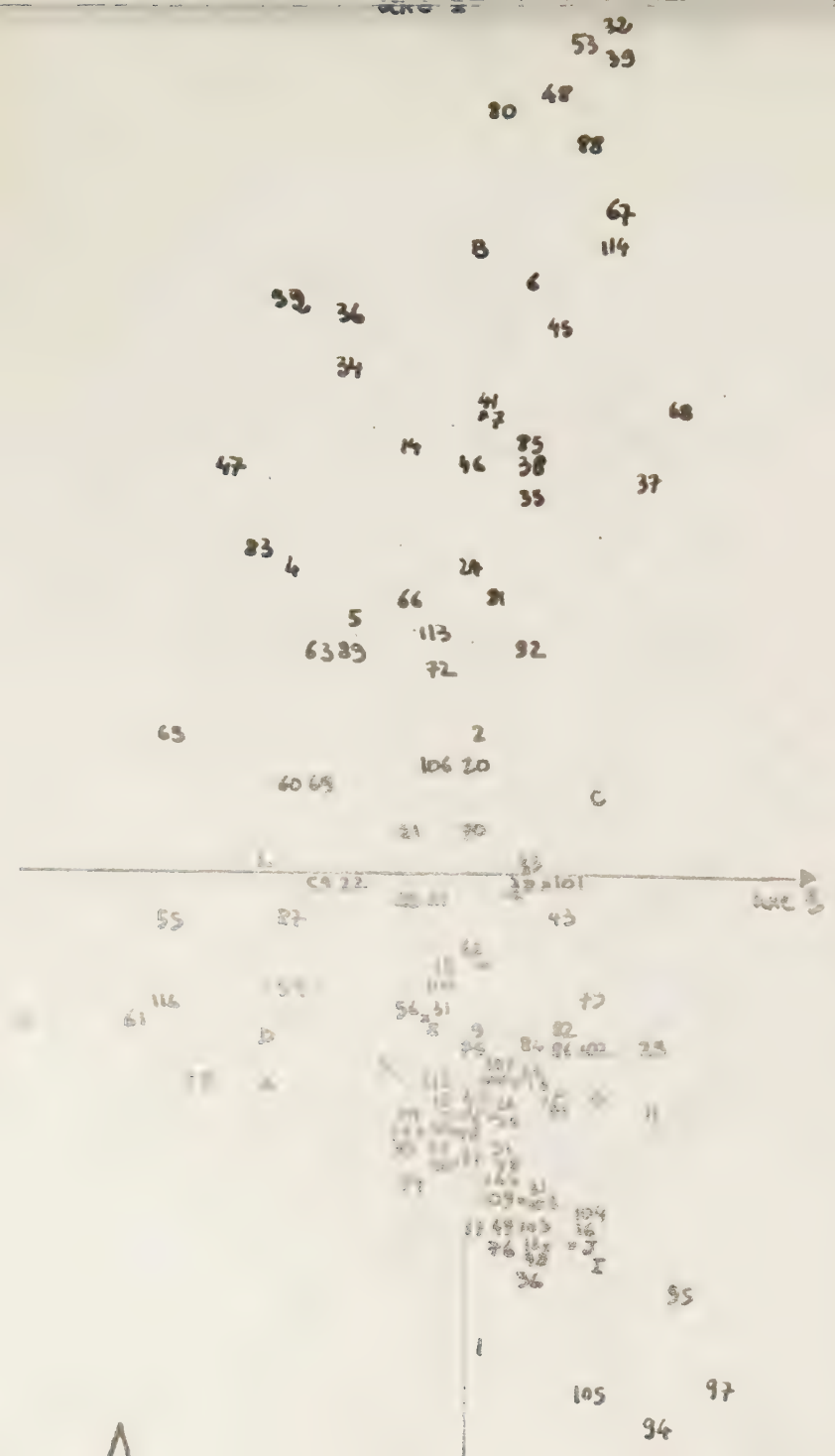

axe 2
Λ_1

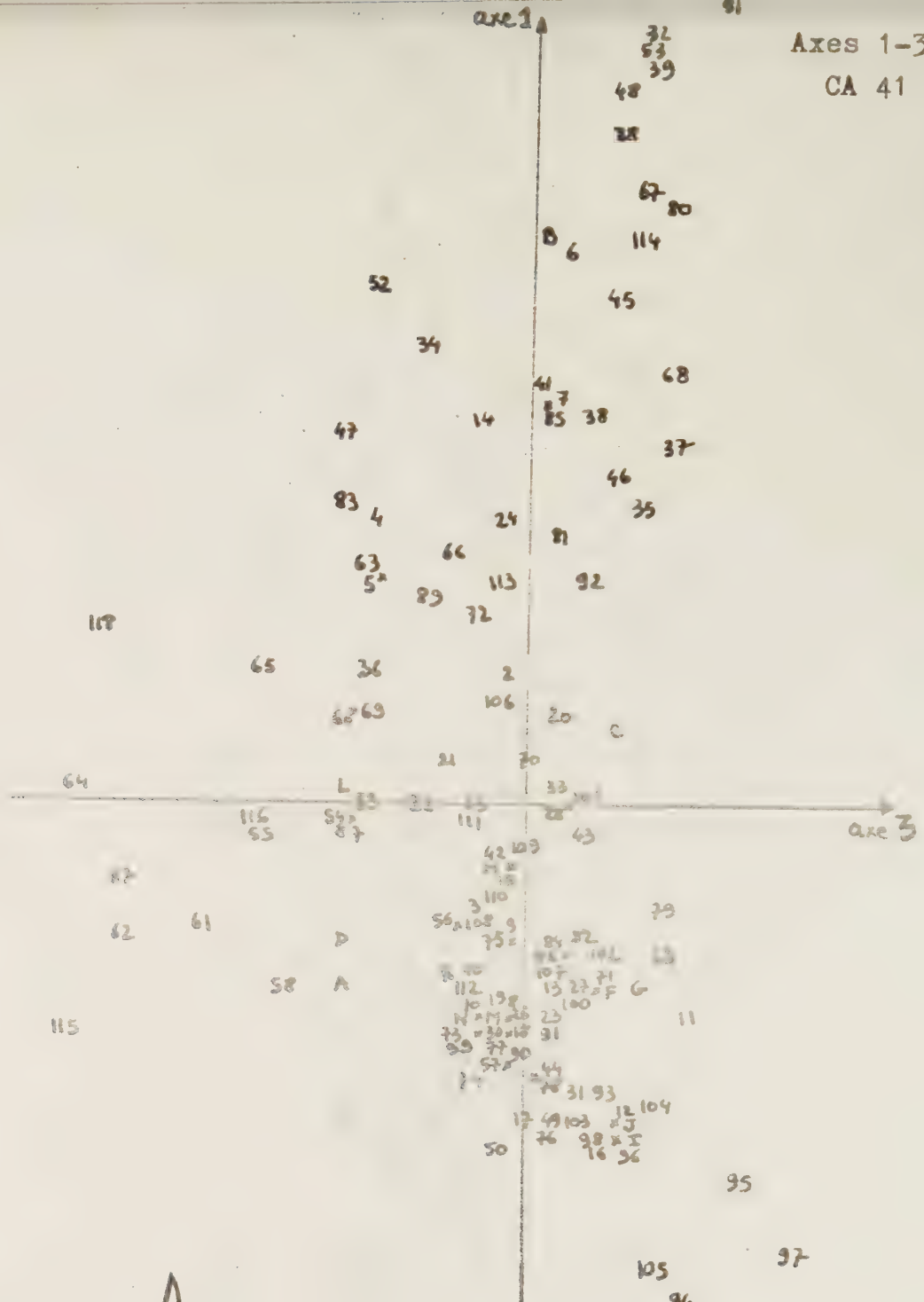

Λ_0


axe 1
axe 3
E

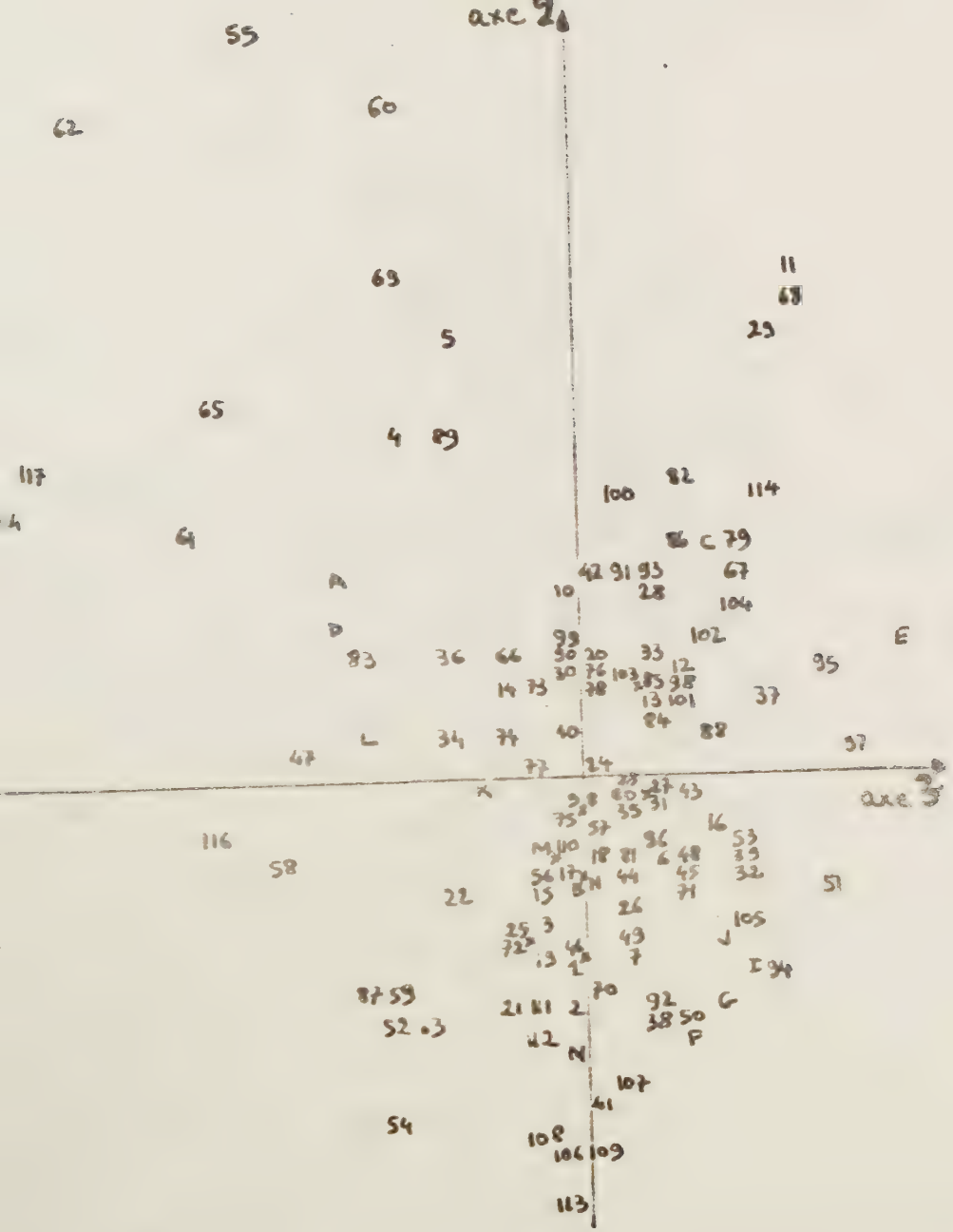

axe 1
axe 2
Axes 1 - 3
CA 41
Trous
E

Axes 1-3
CA 41

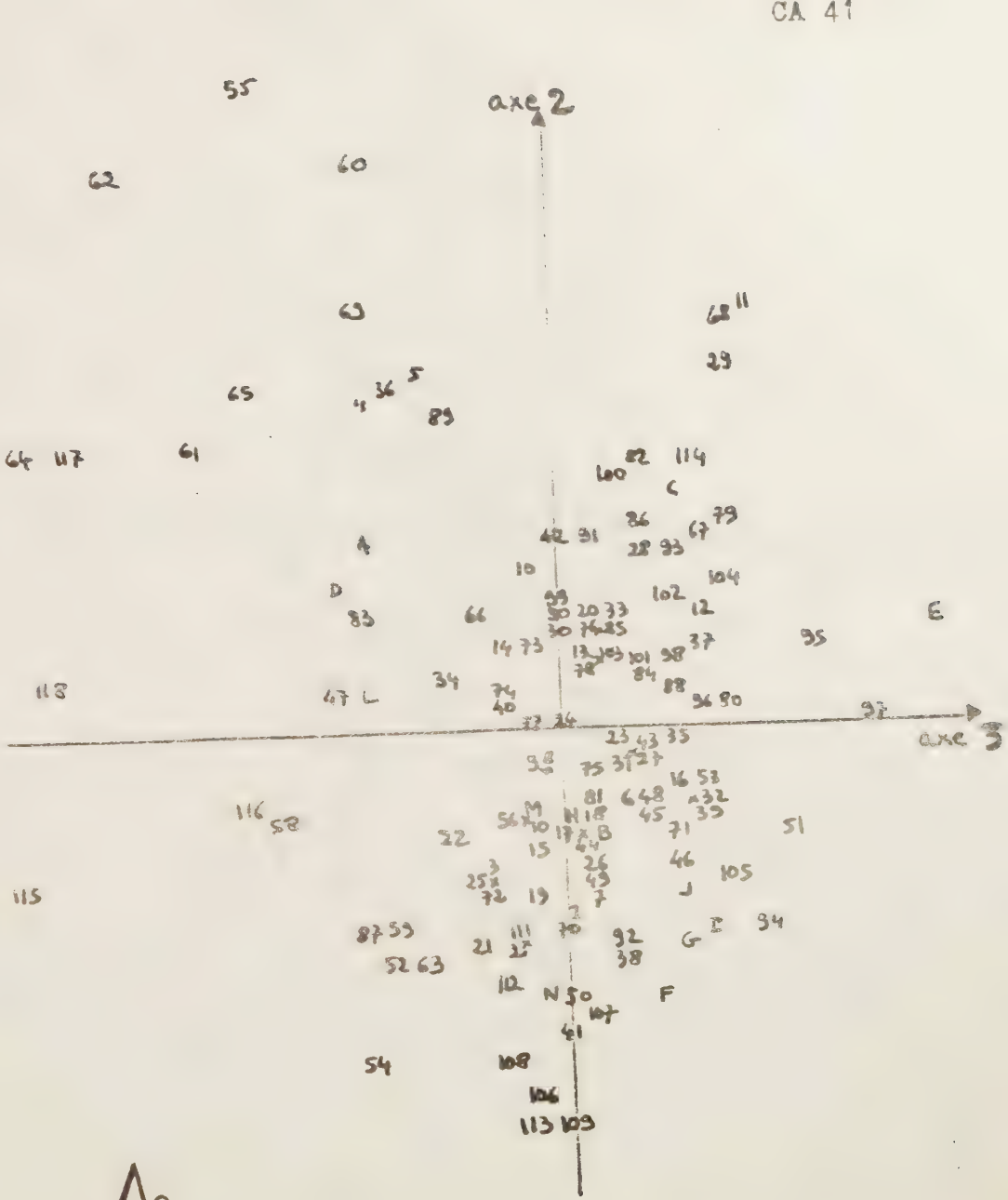




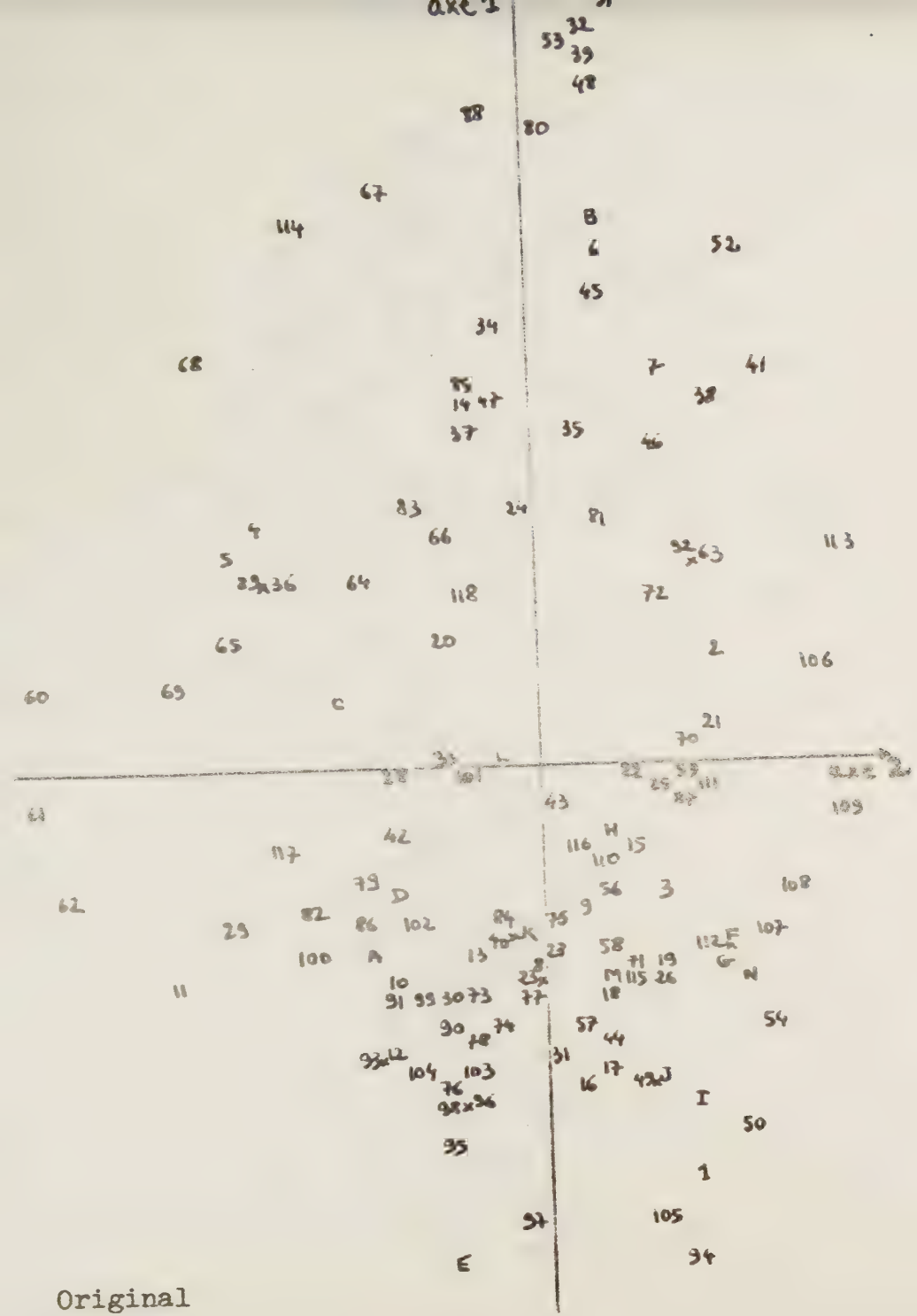
Axes 2 - 3
CA 41



Λ_1



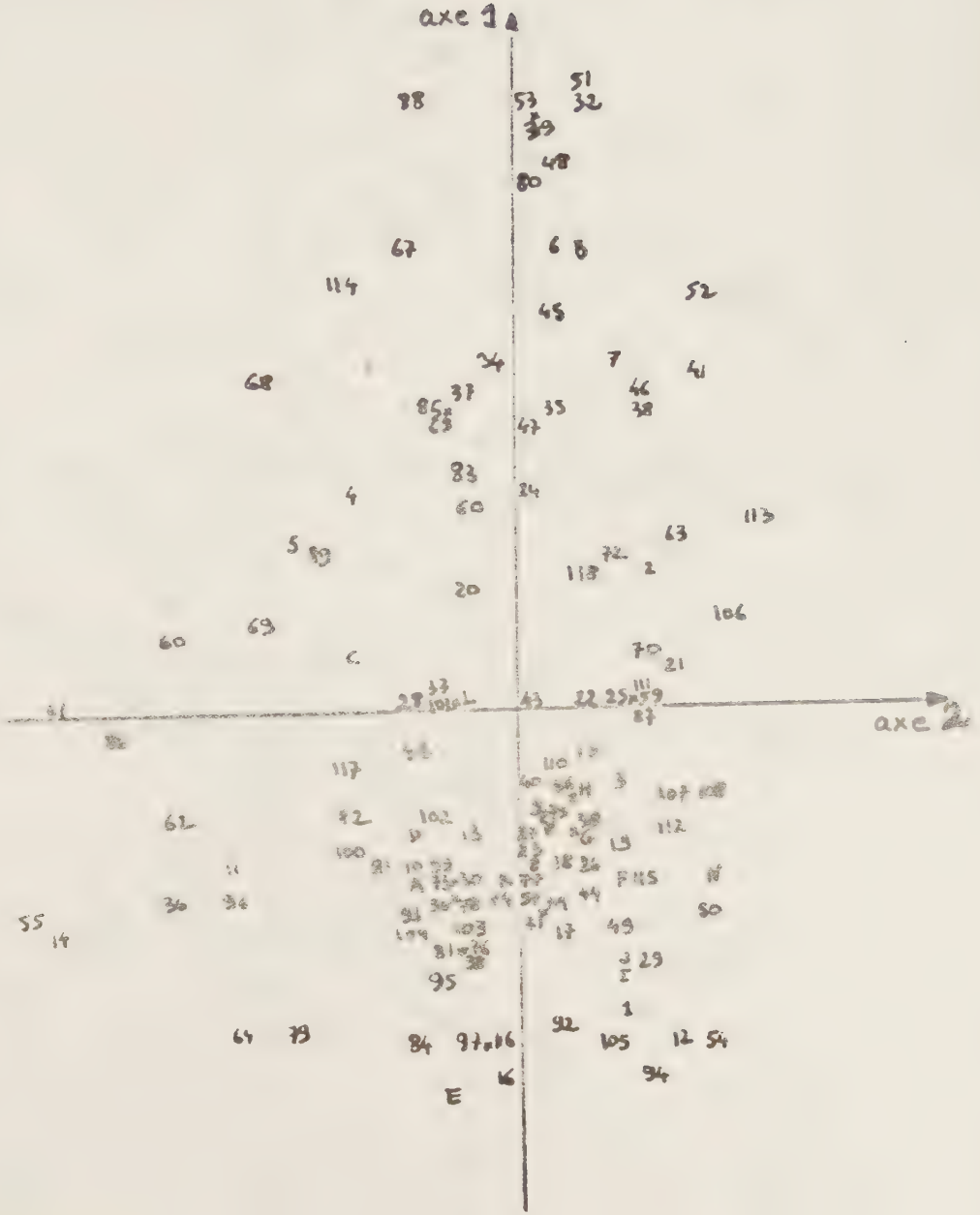
Λ_0



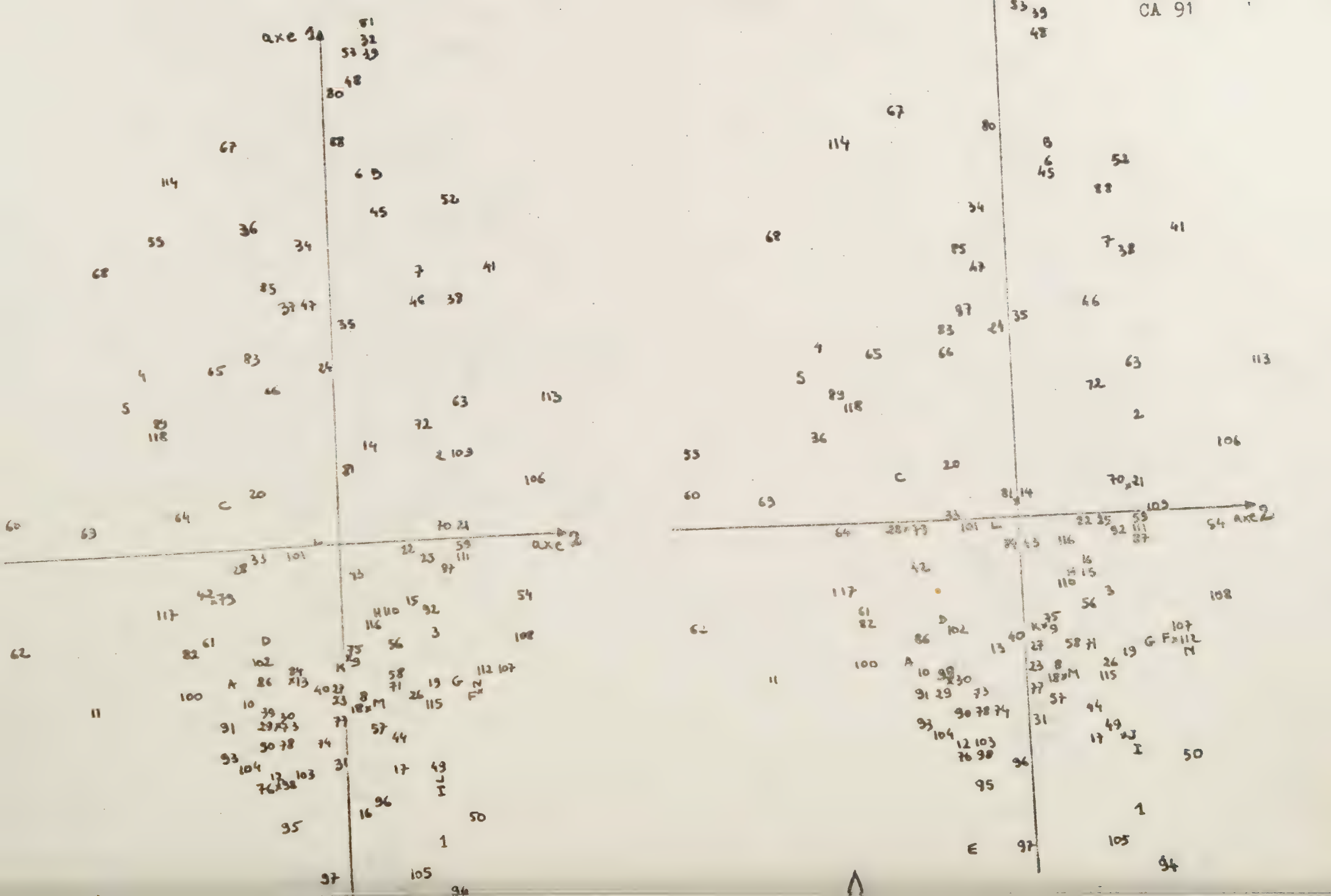
Original

Axes 1 - 2

CA 91

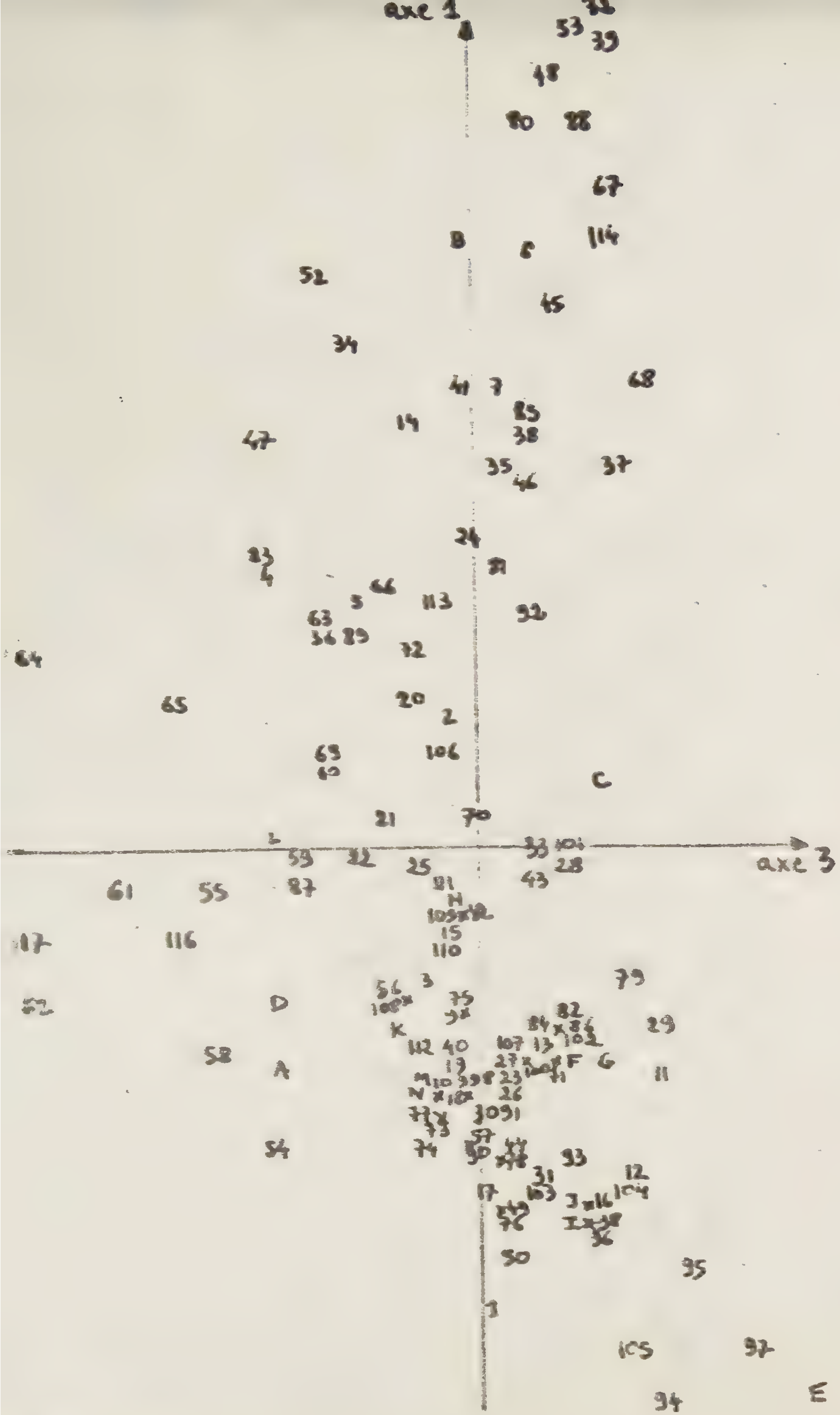


Trous

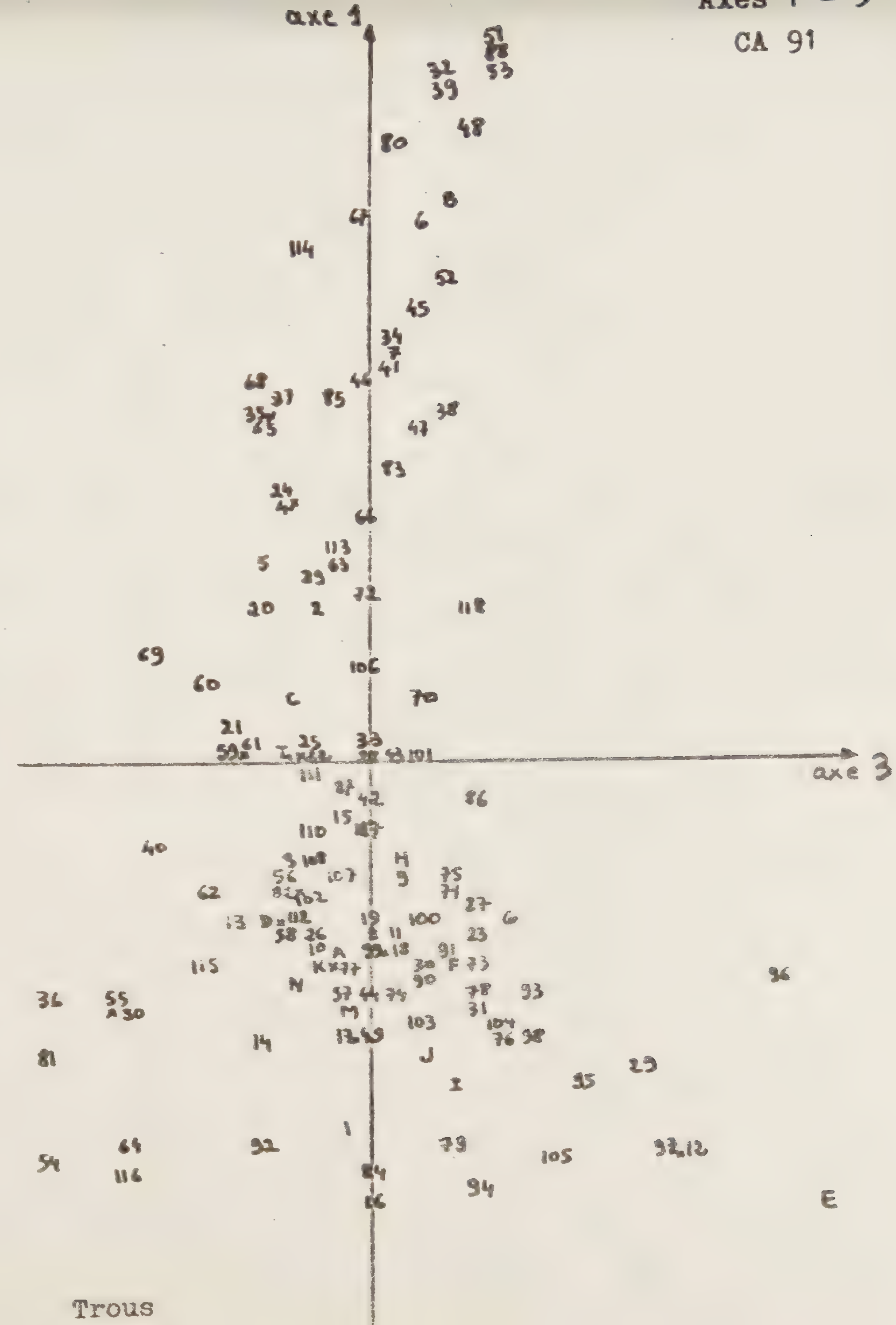

CA 91
axe 1
axe 2

Axes 1-3

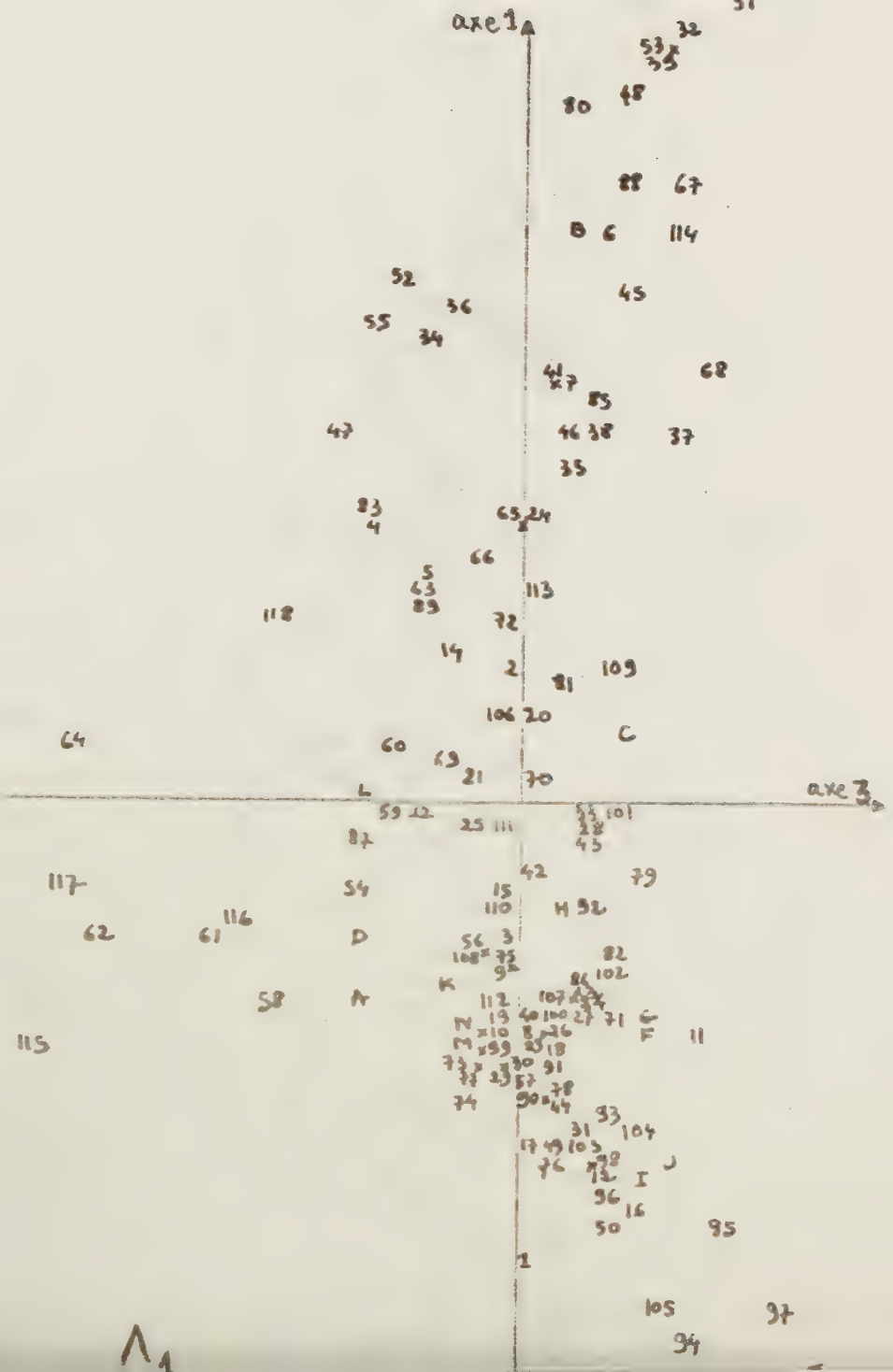
CA 91

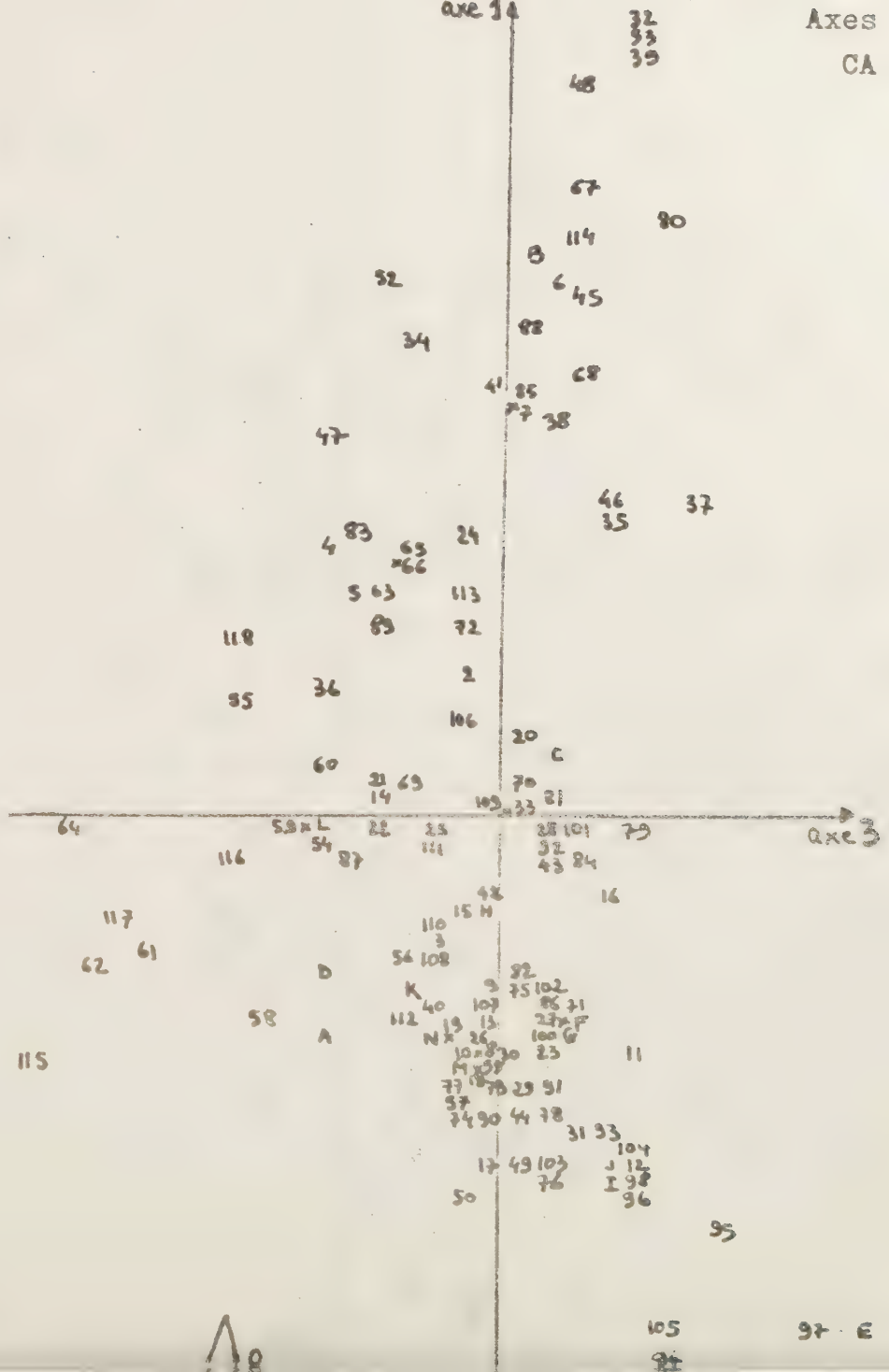


Original

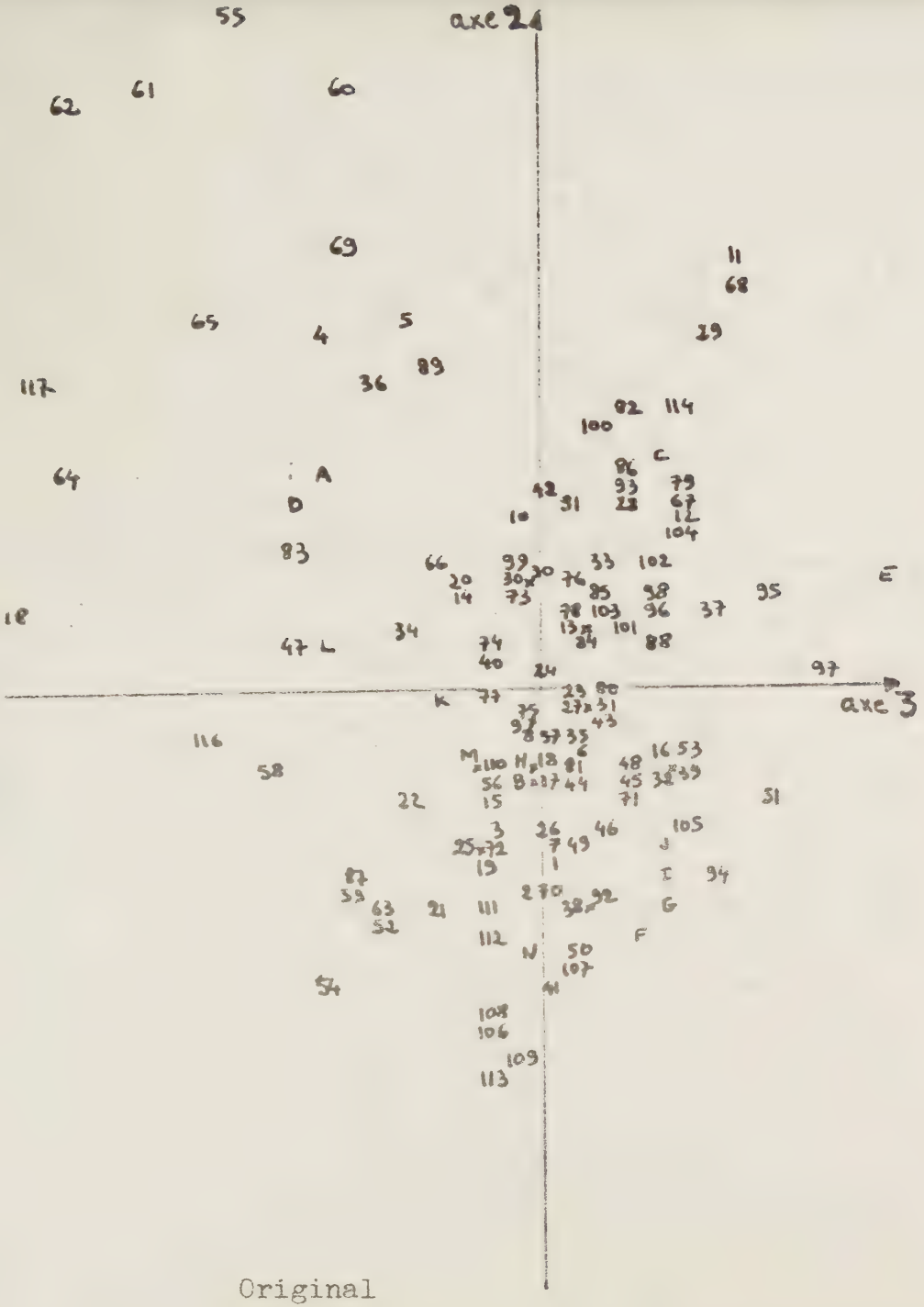


Trous





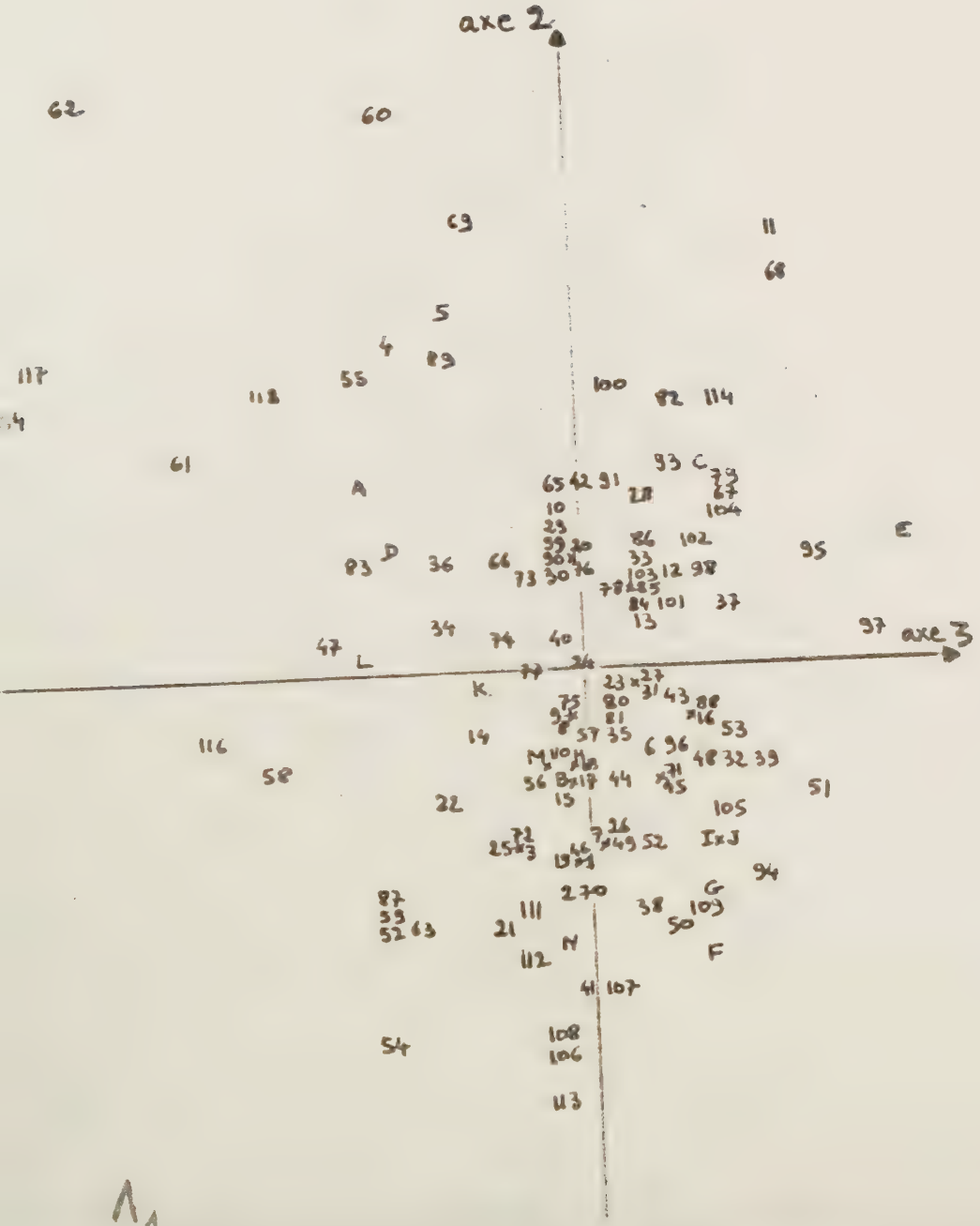
Axes 1 - 3
CA 91



Original

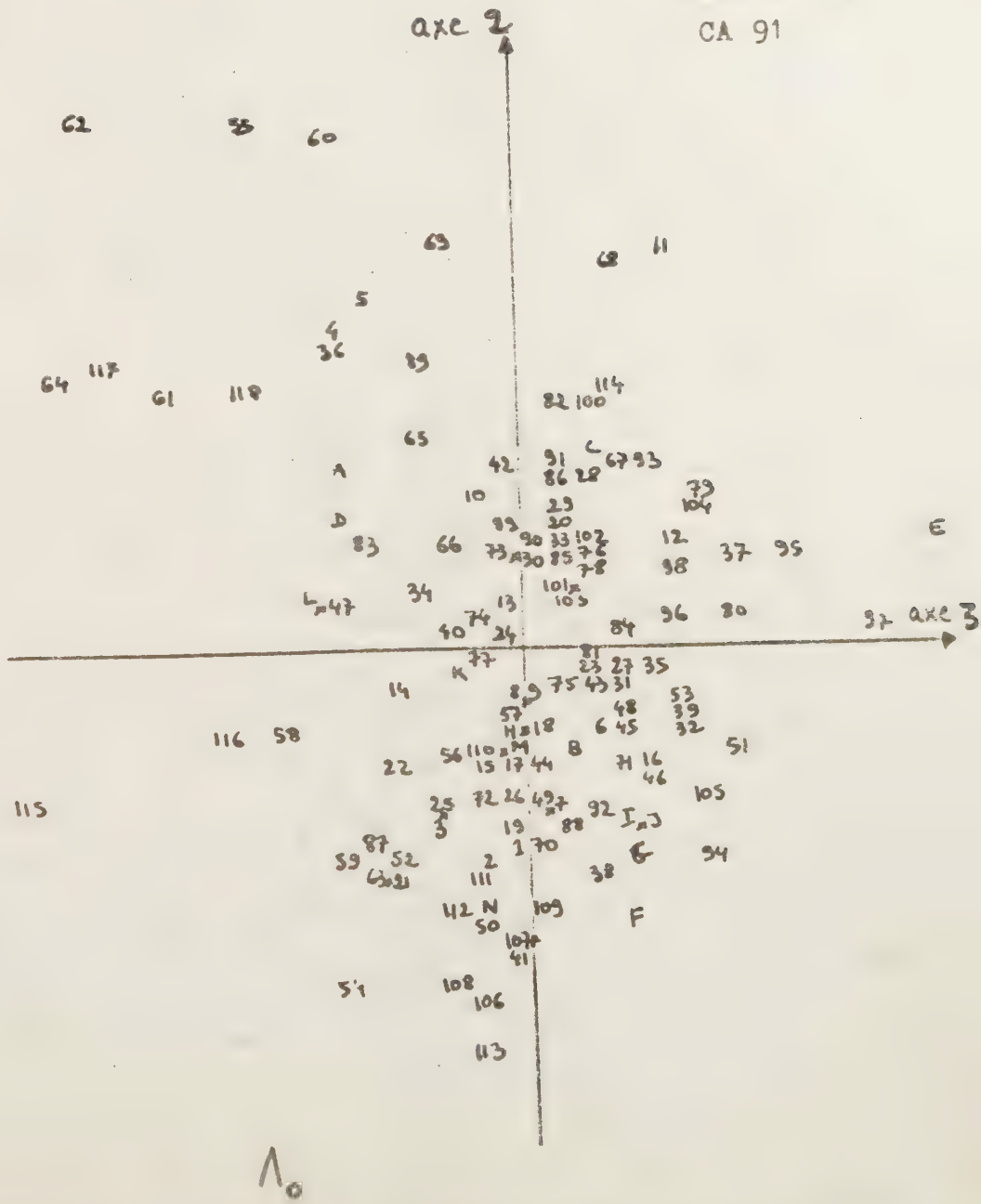
Trous

Axes 2-3
CA 91



Λ_1

Axes 2 - 3
CA 91



Λ_0

La comparaison de ces plans montre que les analyses effectuées en calculant les valeurs des trous par cet algorithme se trouvent très proches des analyses originales.

Pour chaque cas, nous avons comparé les résultats obtenus pour les 3 analyses (avec des zéros, avec les trous remplacés par les nombres calculés dans une approximation de rang 1 et avec ceux d'une approximation de rang soit 2 soit 0) avec ceux obtenus dans l'analyse originale.

Pour cela, nous avons calculé les coefficients de corrélation C entre les facteurs F et les facteurs G ainsi que les nombres donnés par la formule

$$v = \frac{\sqrt{\Sigma\ (F-f)^2}}{\sqrt{\Sigma\ (f-\bar{f})^2}}$$

F étant un facteur de l'analyse à étudier

f le facteur correspondant pour l'analyse originale

$\bar{f}$ la moyenne de f

Une analyse sera d'autant plus proche de l'analyse originale que C_I, C_J seront proches de 1 et v_I,v_J de 0.

Les tableaux suivants donnent ces résultats et les différentes valeurs propres des tableaux comparés avec leurs pourcentages d'inertie.

On voit qu'en général, les coefficients de corrélation sont très élevés pour les tableaux remplis en utilisant l'algorithme.

En faisant pour r = 1 comme pour r = 0 nous pouvons comparer les valeurs obtenues pour un même trou dans les différents cas étudiés.

MU

Valeurs propres

			1	2	3	4	5
Original	λ		0,00119	0,00019	0,0001	0,000023	0,000015
	test	6,81	1,55	0,72	0,26	0,16	0,0098
	%		77,27	12,22	6,62	1,47	0,97
Trous	λ		0,048	0,026	0,014	0,0094	0,0062
	test	434	241	138	83	45	21
	%		44,47	23,72	12,65	8,69	5,73
Λ_o	λ		0,0001	0,00022	0,00011	0,000023	0,000017
	test	6,16	1,75	0,77	0,29	0,19	0,11
	%		71,51	15,92	7,85	1,67	1,22
Λ_1	λ		0,0015	0,00018	0,000053	0,000023	0,000016
	test	8,19	1,33	0,52	0,28	0,179	0,11
	%		83,52	10,07	2,92	1,28	0,87

Corrélations

Trou-Original	ϱ_I	0,87	0,61	0,29	0,35	0,66
	ν_I	5,7	11,6	12,5	21	21,4
	ϱ_J	0,0004	0,14	0,23	0,8	0,12
	ν_J	7,1	7,8	11,1	25,5	16,4
Λ_o-Original	ϱ_I	0,95	0,93	0,91	0,98	0,89
	ν_I	0,3	0,4	0,4	0,2	2
	ϱ_J	0,98	0,97	0,79	0,96	0,92
	ν_J	0,2	0,3	0,9	0,3	2
Λ_1-Original	ϱ_I	0,95	0,92	0,87	0,93	0,88
	ν_I	0,4	0,4	1,7	0,4	0,5
	ϱ_J	0,99	0,96	0,71	0,89	0,91
	ν_J	0,2	0,3	1,8	0,5	0,4

CI 9

Valeurs propres

			1	2	3	4	5
Original	λ		0,179	0,034	0,0085	0,0075	0,007
	test	2940	860	469	370	282	201
	%		70,74	13,3	3,37	2,98	2,76
Trous	λ		0,17	0,032	0,013	0,011	0,0095
	test	3060	1091	725	575	444	335
	%		64,36	11,98	4,88	4,29	3,56
Λ_1	λ		0,178	0,033	0,0084	0,0073	0,0067
	test	2910	839	456	359	274	196
	%		71,21	13,14	3,34	2,91	2,66
Λ_2	λ		0,178	0,033	0,0084	0,0073	0,0067
	test	2910	840	451	354	269	192
	%		71,14	13,35	3,34	2,92	2,67

Corrélations

Trou-Original	$\mathcal{C}_I$	0,9979	0,986	0,058	0,02	0,22
	v_I	0,07	0,16	1,93	1,79	1,71
	$\mathcal{C}_J$	0,9988	0,983	0,376	0,23	0,05
	v_J	0,05	0,18	1,93	1,79	1,71
Λ_1-Original	$\mathcal{C}_I$	0,99975	0,9981	0,91	0,89	0,94
	v_I	0,02	0,06	0,43	0,46	0,33
	$\mathcal{C}_J$	0,99972	0,9985	0,87	0,90	0,93
	v_J	0,02	0,06	0,5	0,46	0,37
Λ_2-Original	$\mathcal{C}_I$	0,99982	0,9989	0,89	0,87	0,96
	v_I	0,02	0,05	0,5	0,5	0,3
	$\mathcal{C}_J$	0,99988	0,9983	0,84	0,90	0,96
	v_J	0,02	0,06	0,6	0,5	0,3

CI 23

Valeurs propres

			1	2	3	4	5
Original	λ	2940	0,179	0,034	0,0085	0,0075	0,007
	test		860	469	370	282	201
	%		70,74	13,3	3,37	2,98	2,76
Trous	λ	3410	0,178	0,037	0,022	0,017	0,015
	test		1442	1038	797	611	450
	%		57,71	11,85	7,06	5,46	4,71
Λ_1	λ	2850	0,176	0,032	0,0075	0,0073	0,006
	test		814	440	353	267	197
	%		71,5	13,1	3,04	2,99	2,45
Λ_2	λ	2910	0,179	0,035	0,0074	0,0069	0,006
	test		827	426	340	260	193
	%		71,57	13,78	2,94	2,77	2,3

Corrélations

Trou-Original	$\mathcal{C}_I$	0,9896	0,91	0,09	0,39	0,15
	v_I	0,01	0,4	2	1,4	1,9
	$\mathcal{C}_J$	0,997	0,96	0,31	0,51	0,31
	v_J	0,08	0,3	2,1	1,1	2,1
Λ_1-Original	$\mathcal{C}_I$	0,9995	0,995	0,82	0,76	0,68
	v_I	0,03	0,1	1,8	0,7	1,7
	$\mathcal{C}_J$	0,9995	0,997	0,80	0,77	0,51
	v_J	0,3	0,08	1,9	0,7	1,6
Λ_2-Original	$\mathcal{C}_I$	0,999	0,998	0,24	0,44	0,44
	v_I	0,04	0,09	1,6	1	1,6
	$\mathcal{C}_J$	0,9998	0,998	0,29	0,36	0,19
	v_J	0,02	0,07	1,7	1,5	1,5

Valeurs propres

			1	2	3	4	5
Original	λ	2940	0,179	0,034	0,0085	0,0075	0,0057
	test		860	469	370	282	201
	%		70,74	13,3	3,37	2,98	2,76
Trous	λ	4320	0,19	0,055	0,042	0,034	0,021
	test		2375	1813	1384	1037	826
	%		44,99	13,01	9,95	8,02	4,86
Λ_1	λ	2790	0,176	0,0259	0,008	0,0071	0,0058
	test		745	444	351	269	201
	%		73,31	10,8	3,32	2,96	2,4
Λ_2	λ	2900	0,179	0,0346	0,007	0,0068	0,0054
	test		804	400	317	238	174
	%		72,25	13,94	2,86	2,74	2,19

Corrélations

Trous-Original	ε_I	0,96	0,56	0,18	0,2	0,24
	v_I	0,3	1,1	2,4	2,5	2,2
	ε_J	0,992	0,69	0,27	0,31	0,43
	v_J	0,13	0,9	2,2	2,1	1,4
Λ_1-Original	ε_I	0,9991	0,989	0,66	0,64	0,56
	v_I	0,05	0,2	0,8	0,9	1,7
	ε_J	0,9989	0,992	0,49	0,69	0,44
	v_J	0,05	0,2	1	0,8	1,7
Λ_2-Original	ε_I	0,998	0,995	0,31	0,43	0,35
	v_I	0,06	0,11	1,6	1	1,5
	ε_J	0,9992	0,994	0,30	0,37	0,04
	v_J	0,04	0,11	1,6	1,1	1,4

CA 11

Valeurs propres

			1	2	3	4	5
Original	λ		0,127	0,034	0,016	0,012	0,0067
	test	41800	15809	8837	5501	3106	1734
	%		62,21	16,67	7,97	5,72	3,28
Trous	λ		0,129	0,034	0,016	0,012	0,0069
	test	42600	16252	9215	5876	3378	1964
	%		61,85	16,52	7,84	5,86	3,32
Λ_1	λ		0,125	0,033	0,016	0,011	0,0066
	test	41500	15590	8784	5437	3401	1737
	%		62,4	16,42	8,07	5,63	3,29
Λ_2	λ		0,126	0,035	0,016	0,011	0,0064
	test	41800	15829	8614	5284	3009	1699
	%		62,11	17,27	7,97	5,45	3,13

Corrélations

Trou-	ρ_I	0,998	0,997	0,9986	0,993	0,987
Original	v_I	0,06	0,08	0,05	0,13	0,16
	ρ_J	0,9998	0,9993	0,994	0,995	0,9987
	v_J	0,2	0,04	0,04	0,12	0,07
Λ_1-Original	ρ_I	0,998	0,988	0,9993	0,992	0,98
	v_I	o,06	0,15	0,04	0,13	0,2
	ρ_J	0,99995	0,9994	0,9990	0,997	0,9995
	v_J	0,01	0,04	0,05	0,09	0,03
Λ_2-Original	ρ_I	0,986	0,9985	0,995	0,95	0,97
	v_I	0,17	0,06	0,1	0,3	0,2
	ρ_J	0,996	0,9989	0,996	0,96	0,989
	v_J	0,04	0,04	0,1	0,4	0,15

CA 41

Valeurs propres

			1	2	3	4	5
Original	λ	41800	0,127	0,034	0,016	0,012	0,0067
	test		15809	8837	5501	3106	1734
	%		62,2	16,67	7,97	5,72	3,28
Trous	λ	46300	0,136	0,037	0,02	0,015	0,0081
	test		18733	18230	7205	4136	2502
	%		59,5	16,22	8,70	6,63	3,53
Λ_c	λ	41000	0,123	0,033	0,016	0,012	0,0066
	test		15648	8811	5441	3072	1717
	%		61,87	16,66	8,21	5,78	3,30
Λ_1	λ	42100	0,13	0,032	0,016	0,011	0,0065
	test		15223	3652	5365	3050	1708
	%		63,82	15,62	7,81	5,50	3,19

Corrélations

Original-Trous	$\mathcal{C}_I$	0,96	0,96	0,71	0,68	0,94
	ν_I	0,3	0,3	0,8	1	0,4
	$\mathcal{C}_J$	0,996	0,993	0,88	0,39	0,990
	ν_J	0,10	0,13	0,6	1,1	0,2
Λ_c-Original	$\mathcal{C}_I$	0,995	0,989	0,992	0,97	0,97
	ν_I	0,10	0,15	0,13	0,2	0,2
	$\mathcal{C}_J$	0,9995	0,998	0,9988	0,98	0,998
	ν_J	0,04	0,07	0,06	0,2	0,07
Λ_1-Original	$\mathcal{C}_I$	0,991	0,98	0,993	0,96	0,97
	ν_I	0,13	0,19	0,12	0,3	0,3
	$\mathcal{C}_J$	0,9998	0,997	0,997	0,991	0,998
	ν_J	0,03	0,09	0,08	0,17	0,07

CA91

Valeurs propres

		1	2	3	4	5
Original λ test %	41880	0,127 15809 62.21	0,034 8837 16.67	0,016 5501 7,97	0,012 3106 5,72	0,0067 1734 3,28
Trous λ test %	51300	0,152 21330 58,41	0,043 12830 16,58	0,022 8488 8,47	0,017 5159 6,49	0,0093 3334 3,56
Λ_0 λ test %	39100	0,117 14879 61,92	0,0308 8531 16,25	0,015 5385 8,05	0,011 3132 5,77	0,0064 1808 3,39
Λ_1 λ test %	40800	0,13 14072 65,53	0,028 8198 14,39	0,014 5235 7,26	0,011 3055 5,34	0,0062 1775 3,14

Corrélations

Trou-Original	$\mathcal{C}_I$ ν_I	0,89 0,5	0,87 0,6	0,52 1	0,43 1,4	0,87 0,6
	$\mathcal{C}_J$ ν_J	0,994 0,15	0,96 0,3	0,81 0,7	0,12 1,4	0,987 0,2
Λ_0-Original	$\mathcal{C}_I$ ν_I	0,98 0,2	0,97 0,2	0,95 0,3	0,86 0,5	0,94 0,3
	$\mathcal{C}_J$ ν_J	0,9991 0,06	0,997 0,09	0,985 0,2	0,92 0,4	0,989 0,15
Λ_1-Original	$\mathcal{C}_I$ ν_I	0,97 0,2	0,95 0,3	0,96 0,3	0,86 0,5	0,94 0,3
	$\mathcal{C}_J$ ν_J	0,9992 0,05	0,993 0,15	0,996 0,12	0,98 0,2	0,992 0,13

i,j	p(i,j)	CI105	CI55	CI23	CI9
1,1	15	25,49	24,62		
11,1	40	26,33	24,94	24,25	24,10
16,1	0,2	0,67	0,58	0,57	
24,1	3	2,37	4,32	4,39	
26,1	1	0	0	0	0
6,2	39	34	37,35	37,62	
15,2	49	40,28	41,61	42,17	
17,3	15	19,75	18,9	18,47	
20,3	45	41,35	42,3		
22,3	33	37,48	37,05	36,87	38,03
8,4	14	20,4	23,19	22,35	
11,4	2	0	0,74		
12,4	4	12,92	14,88		
21,4	44	30,46	29,97	29,37	
22,4	28	22,35	22,08	21,58	
7,5	13	16,24	17,03	17,27	17,25
8,5	11	10,05	9,21		
10,5	10	8,15	6,91		
21,5	5	6,27	5,24	5,57	5,48
12,6	39	44,18	39,03		
15,6	34	33,71	34,42	34,69	
17,6	43	43,95	41,85	42,54	42,18
30,6	29	32,32	32,90		
8,7	32	35,63	37,81		
14,7	23	35,09	35,75		
19,7	39	35,51	35,64		
24,7	41	35,83	36,59		
27,7	37	33,57	34,19		
4,8	12	10,39	12,03		
6,8	24	19,53	20,74		
14,8	3	0	0	0	0
20,8	19	27,23	27,04	28,49	27,82
28,8	24	26,77	26,32		
12,9	8	17,66	12,92		
2,10	40	36,37	38,07	35,69	
3,10	22	24,12	23,83		
10,10	17	19,83	18,28		
12,10	33	47,83	41,97		
14,10	45	70,58	76,65		
22,10	21	27,78	27,38		
26,10	10	12,36	11,70		
30,10	42	30,03	30,88	29	
1,11	22	25,05	22,08		
13,11	47	33,94	32,95		
14,11	7	18,84	10,54	8,77	
11,12	1	0,15	0,51		
17,12	4	6,27	5,59		
20,12	21	18,79	19,02	20,08	19,61
29,12	1	6,42	5,05		
13,14	69	39,88	41,58		
18,14	46	41,16	40,54	41,70	
20,14	26	34,71	38,48	40,35	
20,15	71	59,97	61,83		
23,15	49	51,57	49,75		
28,15	54	59,25	60,12		

i,j	P(i,j)	CA191	CA91	CA41	CA11
65,1	310	143,68	120,23		
14,2	335	253,70	248,20		
16,2	180	115,96	130,05		
36,2	380	740,64	800,74	814,93	
54,2	170		376,55	353,40	365,23
55,2	110	489,61	521,69		
64,2	310	244,25	228,55	233,45	
79,2	320	308,47	396,92		
81,2	600	427,75	444,10		
84,2	280	223,02	251,67		
92,2	500	220,75	225,93		
109,2	460	688,36	728,63	181,05	
116,2	190	194,81	171,35	169,26	
12,3	550	319,78	403,33	408,44	
29,3	705	325,21	317,83		
109,3	340	628,46	641,93		
118,3	60	247,28	246,63		
4,4	56	22,54	22,38	22,24	
23,4	45	42,62	42,23		
31,4	38	47,20	46,78	47,21	
52,4	6	33,05	33,09	32,53	
66,4	51	30,45	30,28	30,13	
83,4	41	20,54	18,81	18,61	
90,4	33	41,80	41,15		
93,4	43	37,32	39,34		
98,4	47	43,33	43,01		
20,5	20	46,45	59,4	59,32	
35,5	20	15,18	24,38	25,48	25,19
37,5	45	19,18	31,89		
40,5	95	119,98	119,23	117,78	
46,5	70	13,90	28,95	30,27	
50,5	155	223,33	214,04	210,63	
69,5	20	49,06	51,58		
80,5	30	0	0	0	
81,5	65	63,10	67,88		
41,6	9	9,32	10,19	10,45	10,43
83,6	3	6,35	6,25	6,32	
104,6	12	13,93	16,52	16,33	16,29
113,6	10	10,08	11,09		
3,7	24	21,46	21,36	21,49	
13,7	22	24,06	23,83		
16,7	40	29,68	29,04		
19,7	28	21,56	22,32		
25,7	20	21,68	21,71	21,79	
26,7	33	27,74	27,38	27,63	
36,7	5	12,86	15,28		
50,7	42	35,98	34,72	35,05	
71,7	30	31,15	29,84	30,09	
73,7	29	25,38	24,84	24,99	25,16
83,7	2	10,85	10,05		
89,7	5	9,20	7,69		
111,7	22	21,30	21,32	21,39	
10,8	8	11,18	11,04		
11,8	12	12,10	11,88		
37,8	11	9,12	9,54		
44,8	15	15,03	14,69	14,64	14,49
45,8	17	9,08	9,49		
64,8	5	5,77	5,80	5,87	6,20
104,8	10	12,34	12,82	12,74	
2,9	66	41,03	42,40		
6,9	28	15,40	15,98		

60,9	6	17	27,85		
107,9	83	76,69	77,73	77,78	77,58
7,10	31	29,90	25,61	25,09	25,08
43,10	67	61,16	61,50		
71,10	88	86,27	84,97		
91,10	56	69,58	69,87	70,22	
6,11	2	1,59	1,64		
25,11	4	2,74	2,79		
41,11	2	2,12	2,21	2,26	
42,11	5	2,47	2,49		
53,11	1	1,27	1,47		
100,11	2	2,77	2,79		
110,11	3	2,63	2,68	2,71	
115,11	2	1,66	1,71		
118,11	1	1,34	1,37		
1,12	1	1,12	1,22	1,25	1,24
17,12	1	1,08	1,08		
25,12	3	1,31	1,29		
36,12	1	1,11	1,23		
56,12	1	1,04	1,02	1,05	
96,12	1	1,36	1,31	1,34	
118,12	1	0,65	0,64		
6,13	40	25,70	27,12		
27,13	57	63,64	63,18		
70,13	70	49,66	50,07	50,19	
14,14	65	175,34	180,81		
61,14	65	229,26	244,38	243,57	242,09
86,14	360	465,53	463,82		
88,14	170	314,27	273,86		
96,14	380	602,62	599,10	597,23	

CI 105			CA 191			Représentation d'ordre 1, nb. complémentaires		
6,1	6	6,74	12,1	380	329,08	46,7	23	12,93
15,1	13	10,93	18,1	315	320,81	57,7	36	24,40
21,1	1	0	22,1	300	272,25	[illegible]	[illegible]	18,60
22,1	6	5,52	36,1	315	199,78	65,7	15	[illegible]
23,1	12	5,86	38,1	150	248,27	86,7	14	[illegible]
27,2	55	49,52	51,1	40	183,35	104,7	19	24,66
9,3	45	31,72	53,1	100	145,70	2,8	21	10,43
16,3	2,7	3,11	96,1	380	418,07	4,8	5	7,14
23,3	35	36,35	98,1	365	357,23	7,8	17	7,73
27,3	21	30,30	101,1	280	283,51	13,8	9	12,35
16,4	4,4	1,81	7,2	630	415,96	33,8	20	16,64
3,5	7	9,08	12,2	180	0	51,8	5	8,35
14,5	22	20,24	39,2	999	283,49	70,8	9	10,77
24,5	4	8,63	51,2	999	725,51	78,8	8	12,38
27,5	14	13,59	59,2	320	326,83	95,8	15	17,56
11,6	36	45,12	60,2	180	601,14	106,9	63	42,46
13,6	41	33,19	97,2	150		1,10	117	76,59
16,6	[illegible]	2,99	112,2	300	334,33	6,10	21	18,72
19,6	32	33,34	39,3	510	219,82	18,10	105	66,81
20,6	40	[illegible]	44,3	455	472,65	68,10	32	35,59
3,7	[illegible]	[illegible]	54,3	350	803,48	79,10	49	69,03
22,7	33	37,20	57,3	395	386,44	10,11	4	2,72
11,8	11	[illegible]	[illegible]	255	213,31	22,11	5	2,50
1,9	22	25,34	71,3	450	524,53	31,11	4	3,58
5,9	7	18,92	77,3	380	358,24	34,11	1	1,28
8,9	3	5,89	16,4	57	47,09	81,11	3	2,24
20,9	3	0	25,4	35	37,75	94,11	3	4,22
25,9	1	0	48,4	20	23,94	101,11	3	2,60
16,10	[illegible]	[illegible]	63,4	24	23,63	113,11	3	2,26
[illegible]	[illegible]	[illegible]	74,4	31	39,14	23,12	1	1,22
19,10	[illegible]	[illegible]	108,4	31	40,13	31,12	1	1,31
24,10	32	22,33	111,4	36	36,97	42,12	1	1,02
28,10	23	23,53	53,5	20	0	50,12	1	1,52
29,10	69	44,3	93,5	200	128,05	62,12	1	0,53
6,11	51	46,09	104,5	220	140,48	81,12	1	1,03
8,11	46	59,95	1,6	20	14,77	93,12	1	1,06
23,11	48	53,21	17,6	51	13,69	13,13	70	62,44
27,11	33	33,54	19,6	76	12,22	14,13	45	22,05
29,11	17	32,24	24,6	7	9,49	17,13	71	63,44
9,12	13	13,37	34,6	12	5,58	20,13	90	33,53
15,12	11	10,91	36,6	3	7,79	33,13	39	78,50
16,12	2,9	1,33	38,6	11	9,97	50,13	87	94,96
4,13	16	25,58	42,6	10	11,23	56,13	53	53,59
12,13	7	24,40	78,6	9	14,03	90,13	56	67,28
15,13	16	14,95	85,6	3	8,42	95,13	87	96,13
6,14	[illegible]	37,37	96,6	19	17,85	1,14	660	498,42
17,14	[illegible]	[illegible]	108,6	48	13,51	55,14	50	267,39
29,14	[illegible]	46,67	21,7	[illegible]	17,29	[illegible]	[illegible]	[illegible]
2,15	35	38,71	24,7	14	16,16	83,14	150	216,22
8,15	72	49	41,7	27	15,60	89,14	80	180,79

λ 0,182
% 76,88 %
test [illegible]349

λ 0,137
% 69,62 %
test 12235

(par tour, négatifs)

Comme pour l'approximation de rang 0, il faut remarquer que les valeurs trouvées pour un même trou sont très voisines quel que soit le pourcentage de trous dans l'analyse. Or ceux-ci varient de 0,7 à 11,6 et de 2 à 23,3.

Voici, à titre d'indication , les nombres d'itérations nécessaires

r	1	2	3	4	5
MU - 13	>100				
CI - 9	8	17	11	41	42
CI - 22	14	14	58	68	100
CI - 35	19	26	>100		
CA - 11	65	86	68	>100	
CA - 41	78	95	>100		
CA - 71	>100				

On voit qu'en général ce nombre augmente avec le rang r ainsi qu'avec le nombre de trous.

2- Les tableaux PR

Lorsqu'on se trouve en présence de tableaux de grande dimension (par ex, les praomys 265 x 20) et de données manquantes, la première idée pour supprimer cet inconvénient est d'enlever, si c'est possible, les colonnes correspondant aux trous. Cette méthode étant communément utilisée, il semble bon de comparer l'analyse originale avec :

- l'analyse pour laquelle on a affecté aux trous les valeurs calculées à l'aide de l'algorithme (avec remise à zéro).
- l'analyse pour laquelle les colonnes correspondant aux trous ont été otées.

Ce travail a été effectué pour :

	N	%	X
PR25	25	0,5	241
PR50	50	0,9	220
PR100	100	1,9	182

X étant le nombre de colonnes restant une fois otées les colonnes correspondant aux trous.

Les calculs des coefficients de corrélation ont été effectués en utilisant l'approximation de rang 1 mais, dans ce cas encore, la première valeur propre a un pourcentage d'inertie important 45,72 % et les valeurs propres suivantes sont faibles (3.10^{-4}, 2.10^{-4}...).

Pour les calculs des coefficients de corrélation entre les analyses de tableaux de dimensions différentes, les calculs ont porté sur les valeurs des facteurs aux seuls points communs.

Les résultats apparaissent dans les tableaux suivants .

PR 25 265x20 25 trous $\varepsilon = 10^{-4}$

	r		0	1	2	3	4	5
1	<146 , 1 <	271.	270.969711	269.858073	267.037751	263.513346	264.023728	263.554905
2	<212 , 3 <	48.	48.665580	48.170260	48.331868	48.328173	48.847270	49.084037
3	<252 , 3 <	43.	42.926934	45.054090	44.882614	44.868026	44.935406	45.074449
4	<171 , 4 <	150.	146.591118	146.943989	147.069609	147.454986	149.715198	149.516590
5	< 97 , 7 <	48.	50.058028	48.947375	49.094921	49.305515	49.335155	49.251422
6	<156 , 7 <	47.	51.252730	45.867630	45.397748	45.212061	45.933983	46.871323
7	< 25 , 9 <	47.	42.869766	42.843752	42.774703	42.748759	42.406843	42.443077
8	< 73 , 9 <	52.	52.275791	52.289455	52.432156	52.451091	52.849849	52.941613
9	<131 , 9 <	47.	47.452188	47.466910	47.490468	47.494613	47.837323	47.621600
10	<183 , 9 <	50.	50.979448	50.993741	51.605789	51.624540	53.329219	53.195151
11	<187 , 9 <	49.	48.627941	48.642917	48.408419	48.412322	48.781842	48.792857
12	<164 , 10 <	30.	30.945014	30.937345	29.904471	30.022820	30.029040	29.916840
13	<111 , 11 <	88.	90.154907	89.541585	89.500535	89.528302	89.173823	89.343870
14	<133 , 11 <	79.	77.204729	79.150096	79.118482	79.225200	79.680285	78.643763
15	<170 , 11 <	83.	86.257132	85.914957	85.938935	85.933338	85.304648	85.410585
16	<165 , 13 <	22.	22.152090	21.804438	21.778342	21.822414	21.819090	21.850722
17	<204 , 14 <	55.	57.117702	53.320087	54.641128	54.565790	56.284033	56.583409
18	< 29 , 16 <	65.	65.332182	63.872545	63.647190	63.874915	63.873882	63.818426
19	<104 , 16 <	65.	66.280171	64.957076	65.252885	65.212368	65.284026	65.446639
20	< 44 , 17 <	52.	52.108408	51.447053	51.669553	51.455395	51.582238	51.690975
21	<162 , 18 <	124.	120.048242	119.069211	119.067943	87.135213	89.610048	111.166156
22	<111 , 19 <	142.	149.704254	152.527395	149.595893	154.765196	160.319778	160.971605
23	< 67 , 20 <	25.	23.597399	24.801524	24.683605	24.761281	25.407049	25.593320
24	< 96 , 20 <	24.	25.872233	25.410248	24.984777	24.970611	24.350953	23.714994
25	<124 , 20 <	26.	23.961778	24.500075	24.345363	24.339144	24.683034	24.925835

4 *	1 *	2 *	3 *	4 *	5
VALEUR PROPRE*	0.10092075D-02 *	*	*	*	
TESTO.9520+03*	0.51744158D+03 *	*	*	*	
POURCENTAGE *	45.72 *	*	*	*	
VALEUR PROPRE*	0.10093445D-02 *	0.33019479D-03 *	*	*	
TESTO.9530+03*	0.51761246D+03 *	0.37510150D+03 *	*	*	
POURCENTAGE *	45.70 *	14.95 *	*	*	
VALEUR PROPRE*	0.10137159D-02 *	0.33021090D-03 *	0.17836129D-03 *	*	
TESTO.9620+03*	0.52465726D+03 *	0.38214913D+03 *	0.30517428D+03 *	*	
POURCENTAGE *	45.47 *	14.81 *	8.00 *	*	
VALEUR PROPRE*	0.10130072D-02 *	0.33100143D-03 *	0.17601128D-03 *	0.12435977D-03 *	
TESTO.9610+03*	0.52417717D+03 *	0.38132312D+03 *	0.30535992D+03 *	0.25168856D+03 *	
POURCENTAGE *	45.48 *	14.86 *	7.90 *	5.58 *	
VALEUR PROPRE*	0.10099335D-02 *	0.33116374D-03 *	0.16214893D-03 *	0.12492789D-03 *	0.10634330D-03
TESTO.9550+03*	0.51890409D+03 *	0.37597207D+03 *	0.30598773D+03 *	0.25206819D+03 *	0.20616986D+03
POURCENTAGE *	45.65 *	14.97 *	7.33 *	5.65 *	4.81

PR 50 265x20 50 trous $\varepsilon = 10^{-4}$

			0	1	2	3	4	5
1 < 56 , 1 <	308.	176.066870	311.446538	311.553408	306.513188	767.582970	765.671011
2 <146 , 1 <	271.	270.752617	269.813065	267.060425	263.505561	667.689820	667.822814
3 <144 , 2 <	95.	95.202030	95.189568	85.037505	84.339925	85.466658	84.056067
4 <202 , 3 <	49.	51.425875	50.902719	50.762384	50.755818	50.661139	50.647055
5 <212 , 3 <	48.	48.674153	48.181757	48.325568	48.322924	48.163281	48.152815
6 <252 , 3 <	43.	42.834513	45.050763	44.858945	44.851257	44.820745	44.856098
7 < 32 , 4 <	152.	148.157454	147.603350	146.916732	147.008315	330.409254	259.923365
8 < 49 , 4 <	149.	146.665944	145.750215	144.866642	144.311135	216.657710	215.717140
9 <171 , 4 <	150.	146.501512	146.943646	147.063115	147.510535	333.903215	348.646035
10 <223 , 4 <	142.	142.840753	142.940369	141.039778	140.955688	267.049559	267.607447
11 <237 , 4 <	134.	134.756150	135.184041	133.779299	133.134374	280.815817	280.598542
12 <183 , 5 <	121.	121.188371	119.491201	118.767694	119.659408	118.577005	118.679028
13 < 97 , 7 <	48.	50.057039	48.951059	49.096460	49.308325	48.793156	49.058737
14 <156 , 7 <	47.	51.251718	45.875002	45.356334	45.206341	45.725525	45.500204
15 <119 , 8 <	41.	41.433552	42.381713	42.359938	42.356616	42.506194	42.614721
16 < 25 , 9 <	47.	42.867125	42.844079	42.775658	42.744112	43.058816	42.609370
17 < 73 , 9 <	52.	52.272570	52.283594	52.428561	52.448803	52.356166	52.507823
18 <107 , 9 <	51.	50.453576	50.525642	50.359969	50.366145	50.669033	50.437877
19 <131 , 9 <	47.	47.442265	47.461289	47.486546	47.492257	47.455661	47.491720
20 <183 , 9 <	50.	50.581585	50.943009	51.538331	51.548371	51.654383	51.358911
21 <187 , 9 <	49.	48.624945	48.637246	48.405308	48.412147	48.329826	48.480848
22 < 39 , 10 <	32.	31.847679	31.743641	32.070257	32.157342	31.927526	32.228688
23 <164 , 10 <	30.	30.949215	30.940500	29.924329	30.032169	29.911385	29.862644
24 <258 , 10 <	25.	26.380431	26.582516	26.051416	26.008319	25.549262	25.628188
25 <111 , 11 <	88.	90.175027	89.554787	89.515280	89.539572	91.086573	89.539477
26 <117 , 11 <	89.	89.005010	88.458289	88.455631	88.339712	88.300656	88.315746
27 <133 , 11 <	79.	77.220187	79.144303	79.110144	79.230521	79.151786	79.158099
28 <157 , 11 <	84.	85.022924	84.329577	84.312130	84.314120	84.482666	84.503557
29 <170 , 11 <	83.	86.314411	85.924211	85.951415	85.947841	85.785503	85.801137
30 <236 , 11 <	78.	81.525145	80.237107	82.974419	82.603979	83.173173	82.229835
31 < 32 , 12 <	83.	84.194035	82.649939	82.845401	82.846740	82.561702	82.529350
32 <120 , 13 <	22.	22.484172	22.776529	21.781116	21.819181	21.573109	21.642536
33 <165 , 13 <	22.	22.153523	21.807833	77.521677	77.507576	78.219645	77.952031
34 <160 , 14 <	77.	77.815166	79.001774	54.617791	54.534769	54.492369	55.216244
35 <204 , 14 <	55.	57.119158	53.256636	57.905237	57.866276	57.219044	56.731330
36 <121 , 15 <	57.	57.385724	57.460252	63.662590	63.876038	63.404112	63.845812
37 < 29 , 16 <	65.	65.336603	63.979195	58.576711	58.041866	58.992020	57.764109
38 < 54 , 16 <	57.	59.055757	58.847110	65.255618	65.224441	65.446149	65.377136
39 <104 , 16 <	65.	66.284455	64.965374	63.155165	63.403300	63.845872	64.548333
40 <171 , 16 <	63.	62.314746	63.057982	57.360571	58.034279	57.125373	58.402269
41 < 8 , 17 <	60.	55.547651	57.253589	51.638734	51.409491	51.618524	51.142653
42 < 44 , 17 <	52.	52.083585	51.405472	48.762506	48.601561	49.142291	48.887850
43 < 91 , 17 <	53.	50.873902	48.954937	119.070088	86.827274	117.912607	86.171148
44 <162 , 18 <	124.	120.045769	119.060679	149.603580	155.146413	241.090500	156.592591
45 <111 , 19 <	142.	149.742675	152.622091	137.287476	153.301814	136.953617	152.960098
46 <229 , 19 <	137.	145.836053	151.006171	24.683909	24.755699	24.850594	24.868379
47 < 67 , 20 <	25.	23.555311	24.757755	24.584425	24.550273	24.587418	24.976668
48 < 96 , 20 <	24.	25.869544	25.412275	24.344964	24.333158	24.352911	24.403308
49 <124 , 20 <	26.	23.899663	24.497991	23.925087	24.225231	23.254325	23.718279
50 <146 , 20 <	24.	23.543292	23.649518				

	1	2	3	4	5
VALEUR PROPRE	0.101251200-02				
TESTO.5520+C3#	C.519224680+C3				
POURCENTAGE	45.89				
VALEUR PROPRE		0.374047760+03			
TESTO.5520+C3#	C.516555570+C3				
POURCENTAGE	45.77	14.96			
VALEUR PROPRE	0.1C16E8230-02	0.327529300-03	0.179432230-03		
TESTO.5620+C3#	C.522840E170+C3	0.381488900+03	C.304051280+03		
POURCENTAGE	45.63	14.70	8.05		
VALEUR PROPRE	0.234741370-02	0.165015800-02	0.905348460-03	0.395999000-03	
TESTO.2620+C4#	C.1EC367550+C4	0.888501500+03	C.496124350+03	C.324498600+03	
POURCENTAGE	38.82	27.29	14.97	6.55	
VALEUR PROPRE	0.226544570-02	0.152754820-02	0.881486310-03	0.312485050-03	0.165011620-03
TESTO.2490+C4#	C.15C6E5C70+C4	0.845064780+03	C.463174900+03	0.327795350+03	0.256306730+03
POURCENTAGE	39.44	26.60	15.35	5.44	2.87

PR 100 265x20 100 trous $\varepsilon = 10^{-4}$

	r		0	1	2	3	4	5
	< 56 , 1 <	308.	306.060021	311.443812	311.625390	306.646724	306.317731	306.480103
	<146 , 1 <	271.	270.747932	269.314576	267.193745	263.448034	264.118027	263.680663
	< 18 , 2 <	101.	99.802333	103.781130	101.483819	102.660651	101.784549	101.580533
	<135 , 2 <	112.	103.219797	108.111445	109.249433	109.688511	109.459749	109.400656
	<144 , 2 <	95.	94.992582	95.185728	95.114234	94.307378	95.468767	95.358393
	<159 , 2 <	100.	97.524039	98.583640	97.072268	96.763359	96.846044	90.984526
	<208 , 2 <	90.	93.410427	90.195595	90.110012	89.128326	89.186801	89.041246
	<221 , 2 <	99.	99.263225	101.497167	96.424062	97.737409	98.089221	78.332112
	<202 , 3 <	49.	51.424897	50.899707	50.763458	50.787797	50.977249	50.848359
	<212 , 3 <	48.	48.673258	48.179795	48.323428	48.308219	48.815376	49.099150
	<252 , 3 <	43.	42.833690	45.057447	44.907631	44.959622	45.034833	45.199456
	< 32 , 4 <	152.	148.132014	147.583006	146.943863	147.035493	145.512506	140.750534
	< 49 , 4 <	149.	146.640915	145.732891	144.890108	144.281445	140.545761	145.651394
	<171 , 4 <	150.	146.473341	146.912003	147.053857	147.575139	143.806803	149.363517
	<220 , 4 <	142.	137.446785	137.499235	136.965605	136.902432	138.682523	137.441534
	<237 , 4 <	134.	134.772319	135.157211	133.801716	133.068574	132.005338	132.378861
	<183 , 5 <	121.	121.187805	119.489932	118.716789	118.570747	120.327341	120.761479
	<114 , 6 <	48.	48.893894	49.255223	49.251180	49.233088	49.250966	48.814753
	< 69 , 7 <	48.	50.735576	47.260186	46.930047	47.033526	47.178543	47.738822
	< 97 , 7 <	48.	50.078224	48.950852	49.070771	49.367291	49.389493	49.312625
	<156 , 7 <	47.	51.273409	45.863242	45.414456	45.174663	45.916994	46.828546
	<158 , 7 <	43.	45.596283	44.373303	44.319758	44.635902	44.967047	43.862350
	< 85 , 8 <	38.	43.636526	39.490549	39.448243	39.353018	39.124079	39.323087
	<119 , 9 <	41.	41.653076	41.993938	41.077912	40.086385	40.071649	40.590219
	<210 , 8 <	45.	43.399401	43.469693	43.370678	44.513313	44.408080	43.206389
	< 9 , 9 <	43.	47.060460	47.059352	47.184525	47.235177	47.319022	47.324930
	< 25 , 9 <	47.	42.897811	42.870022	42.801026	42.881044	42.545595	42.637598
	< 28 , 9 <	50.	49.251614	49.250690	49.336340	49.362321	48.767447	49.797232
	< 73 , 9 <	52.	52.297795	52.305923	52.475372	52.454322	52.869651	53.019061
	<107 , 9 <	51.	50.518343	50.542208	50.397764	50.443317	49.322627	49.174817
	<131 , 9 <	47.	47.472162	47.481073	47.513094	47.514322	47.899585	47.568119
	<154 , 9 <	42.	45.089596	45.068503	44.839535	44.847575	43.643376	43.769474
	<183 , 9 <	50.	51.006571	50.964407	51.660137	51.650929	53.531401	53.312236
	<187 , 9 <	49.	43.648410	43.657553	48.391901	48.380828	48.753688	43.731769
	<227 , 9 <	45.	44.606943	44.582548	44.565737	44.644602	45.025740	45.484703
36	< 7 , 10 <	36.	32.512409	32.392329	32.679159	32.875858	32.873621	32.703061
37	< 89 , 10 <	32.	31.824720	31.714668	32.030682	32.165080	32.143881	32.091604
38	<120 , 10 <	33.	30.973844	30.928050	31.270901	31.221498	31.202207	31.331410
39	<164 , 10 <	30.	30.926904	30.917222	29.997401	30.093498	30.059678	29.934820
40	<258 , 10 <	25.	26.361413	26.573403	26.087493	26.039832	26.041012	26.084379
41	<111 , 11 <	88.	90.177284	89.559847	89.515918	89.537073	89.149463	89.387656
42	<117 , 11 <	35.	89.003591	88.457879	88.459187	88.326370	83.771422	97.755201
43	<133 , 11 <	79.	77.216354	79.137321	79.107586	79.243601	78.599144	78.666626
44	<157 , 11 <	84.	85.018703	84.329941	84.317057	84.325619	83.655997	84.876169
45	<170 , 11 <	83.	86.310126	85.923607	85.950427	85.951563	85.306224	85.421036
46	<236 , 11 <	78.	80.907703	80.076723	80.037389	80.339302	79.426267	79.561561
47	< 32 , 12 <	83.	84.201719	82.664241	82.985455	82.637978	81.470715	83.028417
48	<115 , 12 <	79.	81.631036	82.521912	81.848306	81.707277	80.338434	75.729290
49	< 85 , 13 <	22.	22.755757	23.859857	23.903005	23.664378	22.860860	23.970204
50	<120 , 13 <	22.	22.457295	22.762300	22.821459	22.777456	22.811583	22.751607
51	<140 , 13 <	23.	22.549322	22.652623	22.666462	22.596870	22.622648	22.536316
52	<161 , 13 <	23.	23.105207	23.521937	23.577389	23.647314	23.649275	23.713492
53	<165 , 13 <	22.	22.154546	21.912729	21.790994	21.857037	21.949915	21.879510
54	< 31 , 14 <	78.	73.363623	77.173435	72.675559	72.702447	74.088204	73.782467
55	<160 , 14 <	77.	77.783667	73.958734	77.441030	77.371855	79.882274	77.647573
56	<190 , 14 <	76.	71.176384	71.473436	73.024837	73.477731	71.646415	71.399731
57	<204 , 14 <	55.	57.096037	53.304553	54.712267	54.658646	56.170128	56.437750
53	<210 , 14 <	70.	72.576097	72.539839	72.379917	74.316692	73.908753	75.316210
59	< 14 , 15 <	52.	55.026754	55.375334	55.474259	53.919366	56.663899	55.379835
60	< 37 , 15 <	60.	61.039611	60.709716	60.387354	62.732564	62.174159	62.298302
61	< 65 , 15 <	52.	49.669845	50.737403	51.008232	50.772064	48.674529	50.520544
62	<121 , 15 <	57.	57.395455	57.470220	57.905876	57.909891	58.025295	58.592801
63	< 29 , 16 <	65.	65.313490	63.861088	63.658602	63.957659	63.311435	63.825973
64	< 45 , 16 <	62.	62.588804	63.100812	63.275397	63.612005	63.257640	53.307444
65	< 54 , 16 <	57.	59.034865	58.926664	58.563207	57.791367	57.153081	57.185512
65	< 80 , 16 <	60.	56.191715	57.071747	57.375895	56.912669	56.877375	56.963306
67	<104 , 16 <	65.	60.261207	54.945953	65.245634	65.195025	65.284925	65.474838
68	<115 , 16 <	53.	61.073682	51.684104	62.124456	62.190920	62.553430	62.645711
69	<143 , 16 <	64.	65.234513	63.551423	63.941678	63.171539	64.214331	64.504421
70	<171 , 16 <	63.	62.291589	63.072397	63.142380	63.491020	63.196513	63.148246
71	<215 , 16 <	63.	61.364669	61.155971	61.450752	51.813037	61.481137	61.593514
72	<231 , 16 <	63.	63.357678	62.555470	61.930003	62.136626	62.399573	62.515184
73	< 8 , 17 <	60.	55.928451	57.246627	57.301745	57.924159	57.701216	57.725751
74	< 21 , 17 <	55.	52.334251	52.948416	52.969858	52.843621	52.948183	53.235325
75	< 44 , 17 <	52.	52.065525	51.395992	51.637797	51.445554	51.574805	51.650897
76	< 81 , 17 <	53.	50.956242	48.938451	49.745549	48.606939	49.643008	49.457316
77	<220 , 17 <	50.	50.006628	49.270489	49.031500	48.972984	49.534545	49.992223
78	<226 , 17 <	57.	56.062414	57.675953	57.244978	57.522240	58.318056	58.230554
79	<231 , 17 <	55.	54.202762	55.022149	54.006165	54.396932	54.229164	54.189598
80	<100 , 18 <	106.	112.917307	104.450599	104.532377	95.672993	91.137314	101.999202
81	<133 , 18 <	132.	123.949856	124.445400	124.425433	113.219177	114.763826	104.416159
82	<162 , 18 <	124.	120.014950	119.955827	111.064655	90.036833	91.403754	119.003861
83	<174 , 18 <	116.	114.978257	111.488272	111.399772	101.037059	103.121710	105.203792
84	<220 , 1[illegible] <	109.	112.259436	109.783980	109.604585	105.211873	111.817942	120.971383

85 <224 , 18 <	113.	116.709620	118.399936	118.440275	124.414851	123.310198	117.929917
86 <236 , 18 <	127.	115.600580	119.946588	119.903826	125.790757	120.845477	110.412018
87 < 55 , 19 <	133.	137.425552	141.451468	138.380553	136.326810	137.834888	143.792143
88 <111 , 19 <	142.	140.867827	132.774378	149.747165	154.544651	160.455229	160.777672
89 <119 , 19 <	128.	135.631555	135.239359	151.331067	164.171013	159.337656	156.961290
90 <210 , 19 <	125.	141.317972	141.253188	142.885786	126.860732	126.651678	133.317472
91 <229 , 19 <	137.	145.956735	151.155701	137.053871	150.504844	156.474032	155.676410
92 < 9 , 20 <	23.	25.349482	25.151610	25.046849	25.329973	25.300928	25.281563
93 < 67 , 20 <	25.	23.601998	24.793661	24.682551	24.784784	25.445433	25.622265
94 < 96 , 20 <	24.	25.877275	25.418243	25.014489	24.976459	24.317208	23.722805
95 <124 , 20 <	26.	23.906436	24.497142	24.350917	24.356880	24.715868	24.933235
96 <128 , 20 <	26.	27.207191	26.735821	26.477443	26.280883	26.241506	26.317335
97 <146 , 20 <	24.	23.549889	23.651673	23.901202	24.216344	24.459768	24.395426
98 <167 , 20 <	26.	25.893298	25.645122	25.630903	25.373392	25.281053	25.351191
99 <201 , 20 <	26.	26.694452	25.899171	26.043464	26.129045	25.775238	25.284406
100 <246 , 20 <	27.	24.851798	25.252667	25.511596	25.447597	25.611332	25.565183

M	1	2	3	4	5
VALEUR PROPRE	0.10115434D-02				
TEST0.945D+03	0.50849469D+03				
POURCENTAGE	46.20				
VALEUR PROPRE	0.10085629D-02	0.33026054D-03			
TEST0.947D+03	0.51189913D+03	0.36934565D+03			
POURCENTAGE	45.96	15.05			
VALEUR PROPRE					
TEST0.965D+03	0.52417547D+03	0.37835481D+03	0.29960058D+03		
POURCENTAGE	45.70	15.11	8.16		
VALEUR PROPRE	0.10197752D-02	0.33475353D-03	0.17856153D-03	0.12426319D-03	
TEST0.962D+03	0.52198689D+03	0.37751261D+03	0.30044831D+03	0.24681830D+03	
POURCENTAGE	45.75	15.02	8.01	5.57	
VALEUR PROPRE	0.10121197D-02	0.33155017D-03	0.17374247D-03	0.12516747D-03	0.10779766D-0
TEST0.957D+03	0.52054374D+03	0.37744038D+03	0.30244982D+03	0.24842516D+03	0.20189764D+0
POURCENTAGE	45.63	14.95	7.83	5.64	4.86

PR 197 265x20 197 trous

		0	1	2	3	4	5
17 , 1)	312.	306.370345 *	308.552143 *	309.222349 *	311.454288 *	312.273490 *	313.11?214 *
56 , 1)	3.8.	305.927614 *	311.533096 *	311.755315 *	306.713967 *	306.222061 *	306.346118 *
58 , 1)	2.9.	224.464979 *	203.657691 *	206.248413 *	207.218403 *	207.069173 *	207.422480 *
64 , 1)	318.	315.?03614 *	322.484401 *	328.473.97 *	319.134367 *	321.213322 *	320.427177 *
65 , 1)	276.	280.691906 *	275.601141 *	275.860339 *	274.331654 *	275.073911 *	275.346109 *
67 , 1)	265.	273.146302 *	263.012989 *	264.451563 *	263.290807 *	265.375578 *	265.163506 *
84 , 1)	318.	313.311321 *	319.154723 *	320.740383 *	319.640169 *	319.220644 *	319.650803 *
89 , 1)	317.	305.325891 *	309.195229 *	311.428471 *	313.257950 *	314.850050 *	314.416350 *
98 , 1)	304.	299.903742 *	303.978994 *	307.591535 *	306.790652 *	306.631775 *	306.705453 *
143 , 1)	321.	307.?79148 *	312.954964 *	316.445639 *	313.525499 *	314.701676 *	316.022019 *
146 , 1)	271.	270.636719 *	269.698189 *	266.882649 *	263.124743 *	263.994319 *	263.556449 *
164 , 1)	297.	299.501443 *	303.048430 *	293.362354 *	295.363410 *	296.757917 *	296.797000 *
193 , 1)	305.	304.820784 *	303.557701 *	305.933417 *	306.444168 *	303.562192 *	305.053154 *
238 , 1)	319.	303.271223 *	306.344437 *	303.902181 *	309.332513 *	312.861228 *	312.574085 *
18 , 2)	101.	99.789142 *	103.752633 *	101.537777 *	102.717697 *	102.372985 *	101.838573 *
135 , 2)	112.	108.210073 *	108.153114 *	109.245527 *	109.604858 *	109.430704 *	109.412950 *
138 , 2)	102.	101.046753 *	101.491058 *	100.692415 *	101.272205 *	101.020595 *	100.919655 *
143 , 2)	109.	100.833733 *	104.454701 *	107.267120 *	105.808359 *	106.540577 *	107.224254 *
144 , 2)	95.	94.808499 *	95.547078 *	95.675061 *	94.944184 *	95.845613 *	95.909275 *
159 , 2)	100.	97.514851 *	98.587683 *	97.130944 *	96.885909 *	96.908853 *	97.038252 *
208 , 2)	90.	93.401636 *	90.206730 *	90.095022 *	89.067248 *	89.163284 *	89.070613 *
228 , 2)	85.	93.089236 *	87.746339 *	83.911644 *	84.496498 *	84.494701 *	84.635280 *
231 , 2)	99.	99.256590 *	101.471258 *	96.662919 *	97.989379 *	98.355982 *	98.524420 *
87 , 3)	47.	45.173643 *	45.852045 *	45.864032 *	45.874188 *	45.734810 *	45.944087 *
96 , 3)	51.	48.492210 *	47.305998 *	47.335825 *	47.387549 *	47.741533 *	48.337933 *
126 , 3)	46.	46.986370 *	43.747281 *	43.744127 *	43.767992 *	43.896161 *	43.946239 *
202 , 3)	49.	51.393817 *	50.850203 *	50.778448 *	50.718520 *	50.928590 *	50.793701 *
212 , 3)	48.	48.657943 *	48.159398 *	48.224784 *	48.276002 *	48.660830 *	48.972523 *
252 , 3)	43.	42.820213 *	45.135216 *	45.049603 *	45.040301 *	45.081886 *	45.025469 *
29 , 4)	156.	153.315288 *	152.482451 *	151.867244 *	152.204188 *	151.760247 *	152.430779 *
32 , 4)	152.	148.113711 *	147.584035 *	146.904772 *	147.028015 *	145.560160 *	146.476994 *
38 , 4)	151.	148.483747 *	148.135916 *	148.047126 *	148.536111 *	147.899013 *	148.339773 *
49 , 4)	149.	146.620345 *	145.713107 *	144.811007 *	144.456676 *	146.694417 *	145.682078 *
76 , 4)	125.	123.857221 *	125.690111 *	125.743338 *	125.343076 *	126.353595 *	124.145996 *
98 , 4)	146.	147.787782 *	147.710868 *	149.223475 *	149.076113 *	149.207940 *	150.132322 *
171 , 4)	150.	147.299340 *	147.466189 *	147.651363 *	147.614990 *	150.174146 *	148.040184 *
191 , 4)	129.	127.397878 *	129.039554 *	128.943883 *	128.792544 *	130.866088 *	132.078323 *
230 , 4)	142.	137.417636 *	137.515481 *	136.997844 *	136.855606 *	138.846615 *	137.553942 *
237 , 4)	134.	134.753414 *	135.183381 *	133.778895 *	133.319150 *	132.399377 *	132.129767 *
27 , 5)	117.	118.523831 *	117.064905 *	117.481055 *	117.371243 *	117.540734 *	117.613478 *
72 , 5)	123.	129.433006 *	122.776155 *	122.711449 *	122.617112 *	122.266118 *	122.998381 *
96 , 5)	117.	113.761088 *	110.823536 *	110.256787 *	110.171073 *	110.869606 *	112.169282 *
142 , 5)	118.	102.711423 *	119.112892 *	119.115429 *	118.352906 *	117.343245 *	116.194456 *
165 , 5)	115.	113.298693 *	114.397166 *	114.505216 *	114.315293 *	114.406170 *	114.119385 *
183 , 5)	121.	121.216028 *	119.405908 *	118.678633 *	118.423044 *	119.783300 *	120.368295 *
39 , 6)	46.	51.206418 *	51.405771 *	50.604931 *	50.521974 *	48.949595 *	48.779520 *
95 , 6)	49.	51.780075 *	48.996561 *	48.414113 *	48.630251 *	48.894762 *	48.979962 *
114 , 6)	48.	48.941984 *	49.283730 *	49.272014 *	49.261051 *	49.318825 *	48.839864 *
223 , 6)	47.	46.828511 *	47.214459 *	48.027446 *	47.954975 *	47.960284 *	48.015402 *
226 , 6)	46.	50.698561 *	47.896484 *	48.266557 *	48.137414 *	46.553258 *	46.838674 *
69 , 7)	48.	50.751205 *	47.194 67 *	46.875896 *	46.975814 *	47.049216 *	47.714627 *
97 , 7)	48.	50.093651 *	48.916746 *	49.042886 *	49.307184 *	49.340219 *	49.222341 *
107 , 7)	48.	50.140811 *	46.100872 *	46.267057 *	46.650736 *	47.627673 *	46.973430 *
109 , 7)	48.	47.245739 *	47.101595 *	47.397116 *	47.399845 *	48.270468 *	48.370069 *
144 , 7)	43.	46.232271 *	45.664349 *	45.640844 *	45.846050 *	46.987202 *	46.651136 *
156 , 7)	47.	51.289204 *	45.771104 *	45.339249 *	45.128985 *	45.699251 *	46.718473 *
158 , 7)	43.	45.615329 *	44.349448 *	44.294544 *	44.607419 *	44.951497 *	43.805549 *
194 , 7)	44.	46.118439 *	46.233959 *	46.473700 *	46.588851 *	45.960717 *	45.799478 *
233 , 7)	48.	51.438648 *	47.944999 *	47.763244 *	47.939292 *	46.878824 *	48.203738 *
85 , 8)	38.	43.699604 *	39.542 5. *	39.486151 *	39.335966 *	39.145520 *	39.218512 *
105 , 8)	40.	45.878699 *	40.697047 *	40.762264 *	40.986161 *	41.296228 *	40.852959 *
119 , 8)	41.	41.714436 *	41.983665 *	41.108654 *	40.061096 *	40.028634 *	40.420982 *

63	(163 , 8)	44.	42.392649	*	41.619505	*	41.660994	*	41.784087	*	42.341730	*	41.846929
64	(173 , 8)	40.	41.332411	*	41.077200	*	41.061965	*	41.364593	*	41.879201	*	40.761401
65	(181 , 8)	41.	42.774389	*	40.491645	*	40.471998	*	40.592374	*	39.665639	*	38.362672
66	(184 , 8)	37.	46.250919	*	42.022048	*	42.062168	*	41.956058	*	41.807356	*	40.666402
67	(187 , 8)	44.	43.347257	*	40.953216	*	40.887510	*	40.749055	*	40.209618	*	39.879605
68	(210 , 8)	45.	43.463097	*	43.454899	*	43.338958	*	44.446247	*	44.511919	*	43.279828
69	(9 , 9)	43.	47.056671	*	47.057292	*	47.181763	*	47.246780	*	47.427369	*	47.461589
70	(25 , 9)	47.	42.883517	*	42.881256	*	42.822264	*	42.944049	*	42.654427	*	42.844947
71	(28 , 9)	50.	49.246683	*	49.251312	*	49.340654	*	49.378010	*	48.694655	*	48.827955
72	(40 , 9)	50.	48.311810	*	48.317600	*	48.407184	*	48.299042	*	45.971736	*	47.003414
73	(67 , 9)	42.	44.118198	*	43.835741	*	44.030725	*	44.040044	*	42.756308	*	43.169195
74	(68 , 9)	42.	43.338222	*	43.260914	*	43.611344	*	43.689856	*	42.083656	*	42.000170
75	(73 , 9)	52.	52.363853	*	52.355830	*	52.530736	*	52.482051	*	53.066801	*	53.132269
76	(107 , 9)	51.	50.577846	*	50.470410	*	50.324463	*	50.417808	*	49.172630	*	49.203099
77	(131 , 9)	47.	47.467409	*	47.475468	*	47.506788	*	47.505051	*	47.882606	*	47.504278
78	(141 , 9)	53.	50.428811	*	50.423035	*	50.856936	*	50.776388	*	50.432903	*	50.413366
79	(154 , 9)	42.	45.084991	*	45.081713	*	44.866996	*	44.894665	*	43.449562	*	43.704628
80	(183 , 9)	50.	51.002314	*	50.956161	*	51.651114	*	51.578744	*	53.674029	*	53.262534
81	(187 , 9)	49.	48.623854	*	48.560969	*	48.292866	*	48.275841	*	49.777586	*	48.581844
82	(227 , 9)	45.	44.602476	*	44.597947	*	44.597762	*	44.705360	*	45.338612	*	45.817892
83	(7 , 10)	36.	32.507871	*	32.387370	*	32.679306	*	32.862713	*	32.802384	*	32.680878
84	(12 , 10)	30.	29.566501	*	29.663201	*	29.640362	*	29.873316	*	29.777482	*	29.593805
85	(36 , 10)	32.	31.342783	*	31.326375	*	31.089306	*	31.082718	*	31.054200	*	31.042547
86	(42 , 10)	24.	24.295876	*	24.559540	*	24.677905	*	24.508759	*	25.072676	*	25.069715
87	(55 , 10)	30.	30.359391	*	30.285207	*	30.406243	*	30.458507	*	30.409916	*	30.381223
88	(72 , 10)	35.	34.865816	*	34.367233	*	34.493344	*	34.514737	*	34.475989	*	34.583470
89	(89 , 10)	32.	31.572649	*	31.569666	*	31.907307	*	32.067377	*	31.996099	*	31.952264
90	(120 , 10)	33.	30.969460	*	30.924841	*	31.268532	*	31.244649	*	31.163924	*	31.254837
91	(164 , 10)	30.	30.970364	*	30.971264	*	29.873760	*	29.972184	*	29.926933	*	29.854501
92	(258 , 10)	25.	26.357733	*	26.580382	*	26.091088	*	26.034899	*	26.064150	*	26.082910
93	(30 , 11)	88.	88.732468	*	88.230412	*	87.379429	*	87.754697	*	87.529042	*	89.000630
94	(72 , 11)	92.	95.020825	*	92.389440	*	92.428064	*	92.422408	*	91.930349	*	93.010867
95	(73 , 11)	89.	91.364386	*	90.827716	*	90.924297	*	90.780429	*	91.405439	*	90.890362
96	(111 , 11)	98.	90.210357	*	89.576941	*	89.529080	*	89.545893	*	89.217375	*	89.458703
97	(117 , 11)	89.	89.029938	*	88.475417	*	88.470011	*	88.329602	*	89.769578	*	87.765066
98	(133 , 11)	79.	77.241815	*	79.115602	*	79.095158	*	79.233536	*	78.750839	*	78.616184
99	(157 , 11)	84.	84.939385	*	84.306930	*	84.265952	*	84.309003	*	83.589874	*	84.850744
100	(170 , 11)	83.	86.338586	*	85.937439	*	85.964245	*	85.965150	*	85.422329	*	85.439421
101	(201 , 11)	88.	86.376303	*	85.010613	*	84.908060	*	85.012783	*	85.476397	*	87.633659
102	(202 , 11)	89.	88.512866	*	87.980740	*	87.734486	*	87.709908	*	88.082123	*	87.695099
103	(216 , 11)	84.	84.669947	*	84.289991	*	84.383271	*	84.285026	*	84.772409	*	84.082681
104	(229 , 11)	85.	88.223081	*	86.939196	*	86.748990	*	86.739989	*	86.514937	*	86.486183
105	(236 , 11)	78.	80.931408	*	80.101614	*	80.034228	*	80.315844	*	79.291598	*	79.929899
106	(1 , 12)	88.	90.905736	*	89.431827	*	88.914626	*	88.086147	*	86.741491	*	83.450337
107	(23 , 12)	84.	84.595423	*	82.178262	*	82.934770	*	83.651957	*	85.312363	*	88.166903
108	(32 , 12)	83.	84.226955	*	82.676379	*	83.003387	*	82.726681	*	81.822662	*	82.952294
109	(115 , 12)	79.	81.656810	*	82.475817	*	81.774222	*	81.534534	*	80.581435	*	75.336912
110	(151 , 12)	83.	87.539338	*	86.875559	*	86.351182	*	86.263890	*	86.075179	*	85.087786
111	(181 , 12)	87.	83.245663	*	81.277065	*	80.542229	*	80.845520	*	82.110157	*	85.906267
112	(10 , 13)	24.	22.671780	*	23.170512	*	23.164060	*	23.314971	*	23.308281	*	23.374120
113	(85 , 13)	22.	22.750277	*	23.856595	*	23.770049	*	23.892537	*	23.864971	*	23.871429
114	(120 , 13)	22.	22.450977	*	22.770167	*	22.844752	*	22.774253	*	22.847504	*	22.811865
115	(140 , 13)	23.	22.534041	*	22.654818	*	22.672460	*	22.583765	*	22.642275	*	22.595708
116	(161 , 13)	23.	23.009770	*	23.531275	*	23.603750	*	23.675449	*	23.684048	*	23.726867
117	(165 , 13)	22.	22.124939	*	21.814657	*	21.788897	*	21.866639	*	21.866573	*	21.896725
118	(31 , 14)	78.	73.349733	*	73.180954	*	72.642674	*	72.725844	*	73.985808	*	73.832570
119	(91 , 14)	73.	72.711909	*	73.688788	*	72.592535	*	72.855025	*	70.940971	*	72.431572
120	(160 , 14)	77.	77.769941	*	78.994227	*	77.523025	*	77.250835	*	79.264283	*	77.712475
121	(182 , 14)	81.	78.680117	*	80.339855	*	79.630487	*	79.799982	*	78.434593	*	78.689586
122	(190 , 14)	76.	71.162909	*	71.503193	*	72.972329	*	73.632655	*	71.917563	*	71.535963
123	(204 , 14)	55.	57.005227	*	53.217748	*	54.611803	*	54.862073	*	56.302090	*	56.375743
124	(210 , 14)	70.	72.569026	*	72.558101	*	72.368501	*	74.513406	*	74.651513	*	75.246003
125	(14 , 15)	52.	55.012964	*	55.366785	*	55.493759	*	53.669391	*	55.942149	*	55.269338
126	(18 , 15)	59.	58.939934	*	58.469796	*	58.224316	*	59.440442	*	62.728605	*	60.572750
127	(37 , 15)	60.	61.024315	*	60.695427	*	60.687977	*	62.833273	*	62.259983	*	62.132549
128	(66 , 15)	52.	49.765876	*	50.751023	*	51.112360	*	50.662970	*	48.475511	*	50.768218

(77 , 15)	57.	55.543126 *	56.759626 *	56.720923 *	56.753342 *	59.720657 *	58.166769 *	
(121 , 15)	57.	57.381072 *	57.451412 *	57.956396 *	57.658606 *	58.151337 *	58.414522 *	
(123 , 15)	63.	60.149936 *	60.669722 *	60.450245 *	62.817014 *	65.932567 *	58.547641 *	
(221 , 15)	56.	56.239674 *	56.096763 *	56.134406 *	53.717998 *	53.572663 *	53.481075 *	
(29 , 16)	65.	65.213348 *	63.716816 *	63.459871 *	63.711219 *	63.756685 *	63.687047 *	
(45 , 16)	62.	62.594630 *	63.105865 *	63.285417 *	63.603480 *	63.264370 *	63.267114 *	
(54 , 16)	57.	59.046361 *	58.831442 *	58.562936 *	57.821063 *	57.200371 *	57.250493 *	
(80 , 16)	60.	56.196945 *	57.682963 *	57.386200 *	56.955187 *	56.834018 *	57.004866 *	
(104 , 16)	65.	66.267375 *	64.945130 *	65.238964 *	65.225242 *	65.257609 *	65.495593 *	
(115 , 16)	63.	61.089385 *	61.568428 *	62.101609 *	62.219625 *	62.493239 *	62.655046 *	
(143 , 16)	64.	64.366319 *	63.385572 *	63.772948 *	63.523923 *	63.805844 *	64.289664 *	
(171 , 16)	63.	62.656481 *	63.089263 *	63.167745 *	63.272036 *	62.389356 *	61.701490 *	
(215 , 16)	63.	61.379382 *	61.168295 *	61.468704 *	61.816927 *	61.469601 *	61.564549 *	
(231 , 16)	63.	63.361533 *	62.578591 *	61.963064 *	62.133157 *	62.403479 *	62.520327 *	
(265 , 16)	60.	62.673614 *	61.522224 *	61.156981 *	61.606301 *	61.160712 *	60.904145 *	
(8 , 17)	60.	55.887352 *	57.190191 *	57.293041 *	57.958955 *	57.724433 *	57.794583 *	
(15 , 17)	54.	53.134941 *	52.912948 *	52.405186 *	52.572711 *	52.312158 *	52.260982 *	
(21 , 17)	55.	52.295792 *	52.985447 *	52.922426 *	52.740609 *	52.749908 *	53.117717 *	
(44 , 17)	52.	52.027264 *	51.389642 *	51.644792 *	51.402816 *	51.469436 *	51.572103 *	
(66 , 17)	43.	45.977530 *	43.733216 *	44.257503 *	44.210979 *	43.545320 *	43.640319 *	
(68 , 17)	47.	48.077832 *	45.752398 *	46.226435 *	45.734040 *	44.835650 *	45.011307 *	
(81 , 17)	53.	50.818889 *	48.945327 *	48.791583 *	48.617580 *	48.717982 *	48.470083 *	
(83 , 17)	61.	57.263557 *	58.721888 *	59.204893 *	59.009516 *	59.122030 *	59.089624 *	
(89 , 17)	56.	54.709136 *	55.623511 *	56.028430 *	56.347411 *	55.949855 *	55.931670 *	
(105 , 17)	62.	57.091655 *	58.935538 *	59.540531 *	59.221806 *	58.981786 *	59.091190 *	
(157 , 17)	56.	54.005622 *	54.746312 *	54.663998 *	54.641462 *	54.307712 *	54.522645 *	
(173 , 17)	52.	51.434234 *	51.511800 *	51.464932 *	50.946342 *	50.623542 *	50.867162 *	
(220 , 17)	50.	49.966859 *	49.246640 *	49.035221 *	48.993715 *	49.585842 *	49.716670 *	
(226 , 17)	57.	56.179327 *	57.503881 *	57.147761 *	57.428049 *	58.300681 *	58.239717 *	
(231 , 17)	55.	54.162669 *	54.965736 *	53.996323 *	54.408953 *	54.280835 *	54.225556 *	
(35 , 18)	113.	119.405049 *	121.930934 *	121.924202 *	127.640280 *	121.596181 *	102.393757 *	
(100 , 18)	106.	112.561676 *	104.357341 *	104.356399 *	96.902496 *	100.555334 *	103.436996 *	
(109 , 18)	121.	118.716443 *	118.852898 *	118.727809 *	121.836093 *	112.856539 *	112.393027 *	
(130 , 18)	133.	127.349653 *	132.800673 *	132.928037 *	130.597247 *	130.867055 *	120.594722 *	
(136 , 18)	136.	126.641718 *	132.527671 *	132.623237 *	137.056356 *	134.157627 *	135.348650 *	
(138 , 18)	132.	123.813656 *	124.320500 *	124.116926 *	113.559869 *	115.112351 *	103.906302 *	
(162 , 18)	124.	119.955665 *	118.955255 *	118.949948 *	93.030946 *	97.394780 *	123.089765 *	
(171 , 18)	112.	120.266828 *	119.817110 *	118.849179 *	116.882536 *	127.377654 *	151.940546 *	
(174 , 18)	116.	114.921460 *	111.426175 *	111.359470 *	102.696110 *	102.878902 *	105.192074 *	
(220 , 18)	109.	112.198623 *	109.723220 *	109.574690 *	105.123704 *	111.808152 *	123.954636 *	
(221 , 18)	113.	116.673821 *	118.426175 *	118.456627 *	128.378657 *	126.134243 *	116.076578 *	
(224 , 18)	126.	119.877905 *	122.993693 *	122.977435 *	108.996030 *	104.942441 *	107.376728 *	
(228 , 18)	115.	114.063225 *	107.284221 *	106.174124 *	95.045135 *	92.622543 *	103.682652 *	
(236 , 18)	127.	115.545340 *	120.002900 *	120.007377 *	124.756674 *	117.855029 *	102.940958 *	
(243 , 18)	128.	119.955665 *	116.912244 *	116.962405 *	118.827827 *	118.718616 *	98.893324 *	
(30 , 19)	141.	147.473585 *	149.691363 *	121.286379 *	117.076113 *	123.461830 *	127.080092 *	
(42 , 19)	86.	110.048269 *	97.472429 *	93.320519 *	106.106795 *	83.124623 *	80.555500 *	
(47 , 19)	138.	135.438785 *	135.491171 *	138.035239 *	130.993545 *	122.555485 *	121.254085 *	
(51 , 19)	152.	137.015812 *	141.198310 *	174.490405 *	148.201292 *	160.298303 *	162.703680 *	
(55 , 19)	133.	137.512983 *	141.382907 *	137.466181 *	134.554262 *	136.972573 *	141.490341 *	
(111 , 19)	148.	149.929841 *	152.737255 *	149.291672 *	153.828440 *	159.766985 *	160.925317 *	
(119 , 19)	128.	135.690593 *	135.329531 *	150.900907 *	164.036223 *	160.198719 *	158.416157 *	
(137 , 19)	144.	148.147767 *	152.978000 *	145.943237 *	141.807372 *	139.685610 *	139.783055 *	
(198 , 19)	140.	135.253252 *	134.515377 *	118.209034 *	137.512941 *	126.679904 *	125.343902 *	
(206 , 19)	134.	131.635367 *	128.428357 *	139.782542 *	140.961565 *	129.066601 *	131.566667 *	
(210 , 19)	126.	141.378715 *	141.346328 *	143.233767 *	127.343171 *	126.570111 *	131.860036 *	
(229 , 19)	137.	146.294582 *	151.127151 *	136.229651 *	149.742840 *	155.066209 *	154.921400 *	
(9 , 20)	23.	25.380107 *	25.183645 *	25.091893 *	25.368961 *	25.321871 *	25.296138 *	
(67 , 20)	25.	23.795235 *	24.778482 *	24.685732 *	24.797443 *	25.301518 *	25.565023 *	
(96 , 20)	24.	25.816408 *	25.171032 *	24.797633 *	24.775715 *	24.329179 *	23.941132 *	
(124 , 20)	26.	23.935375 *	24.490987 *	24.355664 *	24.369112 *	24.654547 *	24.880255 *	
(128 , 20)	26.	27.240125 *	26.793379 *	26.558051 *	26.387163 *	26.340361 *	26.452190 *	
(141 , 20)	26.	27.198877 *	25.675945 *	26.353352 *	26.064485 *	26.169695 *	26.121478 *	
(146 , 20)	24.	23.576612 *	23.672824 *	23.886480 *	24.169926 *	24.383079 *	24.340864 *	
(167 , 20)	25.	25.924642 *	25.684315 *	25.669179 *	25.420942 *	25.362115 *	25.438804 *	
(169 , 20)	26.	25.876514 *	26.031113 *	25.974105 *	25.958379 *	26.286457 *	26.378914 *	
(184 , 20)	21.	27.982104 *	27.074254 *	26.815774 *	26.676957 *	25.556548 *	26.152626 *	
(201 , 20)	26.	26.700718 *	25.895725 *	26.024273 *	26.097839 *	25.904820 *	25.415779 *	
(246 , 20)	27.	24.881881 *	25.265827 *	25.487786 *	25.415378 *	25.594211 *	25.592675 *	

PR 292 265x20 292 trous

			0		1		2		3		4		5
1	(17 , 1)	312.	306.615390	*	309.250376	*	310.231582	*	312.577509	*	313.864824	*	313.944433
2	(56 , 1)	308.	305.968961	*	311.683810	*	311.852592	*	306.990674	*	305.811665	*	305.918664
3	(58 , 1)	209.	224.495316	*	208.901541	*	206.822314	*	207.444923	*	207.288179	*	207.952659
4	(64 , 1)	318.	315.046187	*	322.609102	*	329.295786	*	317.339684	*	319.937985	*	319.013365
5	(65 , 1)	276.	280.966476	*	275.916747	*	276.175534	*	274.264462	*	275.227138	*	275.128487
6	(67 , 1)	265.	273.182679	*	263.252710	*	264.681894	*	261.871044	*	265.214878	*	264.721660
7	(84 , 1)	316.	313.053625	*	319.282511	*	321.089088	*	320.018407	*	319.126728	*	319.407128
8	(89 , 1)	317.	304.858898	*	308.595017	*	311.184327	*	310.826683	*	314.008285	*	313.235254
9	(95 , 1)	326.	318.941639	*	324.852951	*	329.247860	*	322.867227	*	322.626754	*	322.269788
10	(98 , 1)	304.	299.951701	*	304.005052	*	308.404146	*	307.083120	*	306.672556	*	307.183485
11	(143 , 1)	321.	304.202044	*	307.999310	*	311.937479	*	313.208538	*	320.810260	*	318.494777
12	(146 , 1)	271.	272.371391	*	272.371473	*	268.087019	*	262.189635	*	264.032920	*	263.530047
13	(164 , 1)	297.	299.298984	*	300.084809	*	292.304965	*	294.535952	*	296.603962	*	296.078523
14	(173 , 1)	305.	304.861982	*	303.721598	*	306.567310	*	309.009350	*	304.009276	*	305.638735
15	(226 , 1)	315.	313.224954	*	320.088638	*	317.588390	*	323.329886	*	318.705639	*	319.895179
16	(237 , 1)	264.	275.618583	*	273.518256	*	268.857587	*	262.784058	*	263.505783	*	262.878549
17	(238 , 1)	315.	303.312211	*	306.501976	*	303.752979	*	307.566774	*	313.549803	*	312.362316
18	(18 , 2)	101.	99.798762	*	103.761086	*	101.562996	*	102.965880	*	102.326576	*	101.552254
19	(123 , 2)	109.	101.376608	*	101.259894	*	105.130982	*	107.396558	*	107.015655	*	104.551577
20	(135 , 2)	112.	103.219796	*	108.151957	*	109.271795	*	109.616013	*	109.421448	*	109.333336
21	(136 , 2)	102.	101.039834	*	101.491427	*	100.899981	*	101.023964	*	100.964764	*	100.620261
22	(143 , 2)	109.	99.886793	*	102.009608	*	104.645864	*	105.224014	*	108.156584	*	107.769481
23	(144 , 2)	95.	94.811555	*	95.593945	*	95.824520	*	94.635884	*	96.112911	*	96.271293
24	(146 , 2)	84.	89.434983	*	89.406455	*	86.140661	*	83.388571	*	84.012328	*	83.948781
25	(159 , 2)	100.	97.524038	*	98.629718	*	97.176817	*	97.020414	*	97.001942	*	97.233111
26	(165 , 2)	94.	97.129600	*	96.106743	*	95.361517	*	96.043822	*	95.859414	*	96.026053
27	(208 , 2)	90.	93.410435	*	90.259077	*	90.172740	*	89.256495	*	89.151325	*	88.918223
28	(228 , 2)	85.	93.082862	*	87.788608	*	84.276119	*	84.273665	*	84.375000	*	84.790533
29	(231 , 2)	99.	99.265156	*	101.506971	*	96.808097	*	97.629609	*	98.471878	*	98.896883
30	(41 , 3)	50.	50.569315	*	49.470143	*	49.490252	*	49.469557	*	49.124593	*	49.189399
31	(87 , 3)	47.	45.180483	*	45.856283	*	45.846855	*	45.868333	*	45.789797	*	45.939744
32	(96 , 3)	51.	48.481092	*	47.325038	*	47.296674	*	47.347938	*	47.821669	*	48.290758
33	(126 , 3)	40.	40.941516	*	43.673208	*	43.633238	*	43.676377	*	43.902294	*	43.946286
34	(152 , 3)	45.	46.340643	*	46.744803	*	46.721937	*	46.679743	*	46.788167	*	46.567122
35	(179 , 3)	47.	45.314748	*	47.250780	*	47.222232	*	47.221174	*	46.749720	*	46.593231
36	(202 , 3)	49.	51.380519	*	50.844713	*	50.837372	*	50.765097	*	50.828139	*	50.713595
37	(212 , 3)	48.	48.645366	*	48.159134	*	48.169354	*	48.224462	*	48.694439	*	48.954899
38	(218 , 3)	48.	47.178672	*	47.819217	*	47.809783	*	47.791087	*	47.710837	*	47.460999
39	(221 , 3)	46.	47.479309	*	47.125147	*	47.134650	*	47.018123	*	47.026149	*	46.935827
40	(237 , 3)	46.	45.083990	*	45.553945	*	45.390705	*	45.168091	*	45.190172	*	45.267877
41	(241 , 3)	52.	47.780795	*	49.117226	*	49.095712	*	49.100615	*	49.128000	*	49.084285
42	(252 , 3)	43.	42.809145	*	45.088788	*	45.056046	*	45.023244	*	44.816403	*	44.706240
43	(29 , 4)	156.	153.799589	*	152.781326	*	152.181367	*	152.568372	*	151.589716	*	152.006211
44	(32 , 4)	152.	149.145139	*	147.594536	*	146.959014	*	146.983221	*	145.575048	*	146.340894
45	(38 , 4)	151.	148.512642	*	148.147888	*	148.126508	*	148.801815	*	147.564320	*	148.298191
46	(47 , 4)	146.	142.217420	*	142.237846	*	142.149298	*	142.138232	*	143.322440	*	143.754631
47	(49 , 4)	149.	146.648877	*	145.697222	*	144.826893	*	144.371127	*	146.875964	*	145.816338
48	(76 , 4)	125.	123.891325	*	125.777936	*	125.717714	*	125.128769	*	126.771254	*	124.037154
49	(98 , 4)	148.	147.821254	*	147.730327	*	149.335046	*	149.134105	*	149.350530	*	150.390360
50	(171 , 4)	150.	147.431698	*	147.902789	*	148.157283	*	148.167237	*	150.613511	*	147.724500
51	(183 , 4)	149.	148.218363	*	147.911323	*	149.277572	*	149.246244	*	149.331293	*	151.813140
52	(190 , 4)	147.	146.775269	*	145.584564	*	146.697170	*	146.492298	*	147.585700	*	147.413700
53	(191 , 4)	126.	127.422670	*	129.122892	*	129.027113	*	128.884121	*	130.772503	*	132.222700
54	(230 , 4)	154.	156.798558	*	156.419123	*	154.913554	*	154.535354	*	154.654314	*	152.594600
55	(237 , 4)	134.	135.829483	*	136.098616	*	134.254108	*	132.670831	*	132.686942	*	131.933700
56	(27 , 5)	117.	118.574912	*	117.251283	*	117.459174	*	117.279835	*	117.405327	*	117.524100
57	(72 , 5)	123.	129.276329	*	122.516551	*	122.518777	*	122.573868	*	121.855009	*	122.558300
58	(96 , 5)	117.	113.809149	*	116.951379	*	116.304045	*	116.240463	*	111.069154	*	112.329200
59	(118 , 5)	120.	123.281280	*	120.679763	*	120.487375	*	119.743961	*	121.353222	*	121.143300
60	(142 , 5)	118.	122.764299	*	119.114191	*	119.149876	*	118.593853	*	117.303492	*	116.142900
61	(144 , 5)	111.	116.875082	*	114.099574	*	110.369544	*	110.666807	*	109.654538	*	109.774600
62	(164 , 5)	116.	114.927476	*	114.504664	*	114.891210	*	114.774656	*	114.234045	*	113.775100

(185 , 5)	115.	113.585864 *	114.364823 *	114.437665 *	114.152983 *	114.227037 *	113.964373 *
(183 , 5)	121.	121.143701 *	119.179123 *	118.378397 *	117.985046 *	119.395821 *	119.905541 *
(214 , 5)	112.	116.929069 *	115.590485 *	115.632009 *	115.190388 *	115.332761 *	115.261464 *
(245 , 5)	119.	121.193275 *	119.879906 *	119.653612 *	119.273166 *	118.890889 *	119.935757 *
(4 , 6)	47.	46.461225 *	47.365737 *	47.556267 *	47.435634 *	47.403063 *	46.581013 *
(29 , 6)	46.	50.485393 *	48.140782 *	48.342734 *	48.255211 *	48.565411 *	48.546836 *
(35 , 6)	47.	48.039234 *	46.667727 *	46.788972 *	46.703207 *	47.535830 *	47.182602 *
(39 , 6)	46.	51.234224 *	51.399210 *	50.649054 *	50.442061 *	49.047353 *	48.921315 *
(95 , 6)	49.	51.594967 *	48.971772 *	48.400754 *	48.673053 *	48.667520 *	48.866546 *
(106 , 6)	48.	47.941459 *	46.166636 *	45.899314 *	45.948651 *	46.251674 *	46.447447 *
(114 , 6)	48.	48.968560 *	49.249434 *	49.229029 *	49.225773 *	49.289965 *	48.790681 *
(161 , 6)	48.	49.909113 *	48.760788 *	48.434322 *	48.326488 *	48.293345 *	49.254990 *
(173 , 6)	47.	46.423337 *	46.073344 *	46.124343 *	46.450718 *	46.858265 *	45.424377 *
(223 , 6)	47.	46.853940 *	47.162938 *	47.920484 *	47.904884 *	48.018603 *	47.939472 *
(226 , 6)	46.	50.671183 *	47.952976 *	48.164958 *	47.885226 *	46.727816 *	46.876475 *
(236 , 6)	47.	46.435280 *	43.698249 *	43.798281 *	43.759717 *	44.388325 *	44.083691 *
(31 , 7)	49.	48.913426 *	49.369952 *	49.527533 *	49.518334 *	49.754752 *	49.936023 *
(55 , 7)	43.	47.182860 *	44.909457 *	44.845005 *	44.944784 *	45.135819 *	44.327483 *
(69 , 7)	48.	50.790418 *	47.223768 *	46.846843 *	46.973741 *	46.997316 *	47.661107 *
(97 , 7)	48.	50.132356 *	48.918511 *	49.063226 *	49.318874 *	49.243929 *	49.115461 *
(107 , 7)	48.	50.179521 *	46.117582 *	46.321467 *	46.804390 *	47.513436 *	46.897894 *
(109 , 7)	48.	47.274878 *	47.104661 *	47.416060 *	47.445348 *	48.374396 *	48.415560 *
(143 , 7)	47.	48.749865 *	47.701299 *	47.288191 *	47.201669 *	48.460809 *	48.828754 *
(144 , 7)	49.	46.264347 *	45.586121 *	45.544442 *	45.988104 *	46.795658 *	46.537386 *
(151 , 7)	48.	50.122719 *	49.589235 *	49.113578 *	49.131194 *	49.237221 *	49.612054 *
(156 , 7)	47.	51.328833 *	45.816709 *	45.324308 *	45.173927 *	45.735799 *	46.789007 *
(158 , 7)	43.	45.659452 *	44.362960 *	44.316250 *	44.702735 *	44.990678 *	43.821863 *
(194 , 7)	44.	46.154073 *	46.193218 *	46.718597 *	46.538133 *	46.914599 *	45.761710 *
(233 , 7)	48.	51.478392 *	47.863690 *	47.757889 *	47.784439 *	46.785985 *	48.027622 *
(36 , 8)	42.	43.718873 *	43.466796 *	43.502779 *	43.485852 *	43.828701 *	44.449454 *
(85 , 8)	38.	43.729205 *	39.520477 *	39.603843 *	39.350077 *	39.247531 *	39.263611 *
(102 , 8)	44.	44.001153 *	41.780396 *	41.866182 *	42.229831 *	41.761561 *	43.577579 *
(105 , 8)	40.	45.989734 *	40.711337 *	40.559581 *	40.911934 *	41.115609 *	40.780797 *
(119 , 8)	41.	41.744070 *	41.968987 *	41.546422 *	40.546829 *	40.324998 *	40.401193 *
(163 , 8)	44.	42.423855 *	41.586645 *	41.521263 *	41.774601 *	42.149246 *	41.820370 *
(171 , 8)	42.	43.106426 *	44.629765 *	44.644601 *	44.739528 *	42.955178 *	43.170378 *
(173 , 8)	40.	41.344974 *	40.881962 *	40.886623 *	41.347852 *	41.633889 *	40.127000 *
(181 , 8)	41.	42.804385 *	40.515008 *	40.292291 *	40.396808 *	39.721773 *	38.168293 *
(183 , 8)	42.	45.451440 *	43.560966 *	43.315471 *	43.201189 *	42.339757 *	41.250653 *
(184 , 8)	37.	46.282541 *	42.037385 *	41.941576 *	41.851628 *	41.888360 *	40.536774 *
(187 , 8)	44.	43.376910 *	40.938552 *	40.991042 *	40.763100 *	40.387493 *	39.934632 *
(210 , 8)	45.	43.493677 *	43.436212 *	43.286365 *	44.308415 *	44.255758 *	42.992750 *
(9 , 9)	43.	47.054523 *	47.049877 *	47.090507 *	47.148557 *	47.605027 *	47.728417 *
(25 , 9)	47.	42.881622 *	42.902038 *	42.834480 *	42.861966 *	42.873105 *	43.147359 *
(28 , 9)	50.	49.244507 *	49.251094 *	49.269630 *	49.314118 *	48.764852 *	48.902751 *
(31 , 9)	48.	49.300660 *	49.286146 *	49.225903 *	49.159214 *	48.745052 *	49.670553 *
(37 , 9)	51.	50.813236 *	50.800767 *	50.899437 *	51.023447 *	50.478856 *	50.289106 *
(40 , 9)	50.	48.353466 *	48.362344 *	48.402556 *	48.436972 *	46.720939 *	46.608032 *
(67 , 9)	42.	44.117299 *	43.878042 *	43.882638 *	43.895617 *	42.850519 *	43.317541 *
(68 , 9)	42.	43.335914 *	43.299186 *	43.321994 *	43.455245 *	42.268063 *	42.145459 *
(72 , 9)	57.	54.369600 *	54.045979 *	54.156395 *	54.157683 *	53.390281 *	53.250491 *
(73 , 9)	52.	52.361543 *	52.344188 *	52.412108 *	52.448966 *	53.020170 *	53.212215 *
(87 , 9)	44.	44.605948 *	44.630354 *	44.647687 *	44.706390 *	44.464304 *	44.288340 *
(107 , 9)	51.	50.576778 *	50.438404 *	50.457770 *	50.433219 *	49.181483 *	49.101919 *
(131 , 9)	47.	47.465311 *	47.463242 *	47.491318 *	47.503314 *	47.869406 *	47.457103 *
(141 , 9)	53.	50.444214 *	50.408557 *	50.579225 *	50.736277 *	50.536065 *	50.678019 *
(154 , 9)	42.	45.082990 *	45.106592 *	44.987539 *	44.973544 *	43.412072 *	43.523329 *
(183 , 9)	50.	50.948008 *	50.895153 *	51.009026 *	51.154911 *	53.386434 *	53.442515 *
(187 , 9)	49.	48.622600 *	48.544413 *	48.496735 *	48.411732 *	48.643123 *	48.415857 *
(211 , 9)	49.	48.852481 *	48.853813 *	48.859380 *	48.863777 *	48.505555 *	48.361136 *
(227 , 9)	45.	44.605505 *	44.626855 *	44.558552 *	44.624451 *	45.623685 *	45.301022 *
(229 , 9)	51.	50.428735 *	50.521554 *	50.239589 *	50.497778 *	49.807054 *	49.853941 *
(7 , 10)	36.	32.507423 *	32.393325 *	32.682217 *	32.802604 *	32.764582 *	32.641895 *
(12 , 10)	31.	29.456893 *	29.542100 *	29.525112 *	29.692812 *	29.598488 *	29.492125 *
(36 , 10)	32.	31.375181 *	31.367886 *	31.149747 *	31.115878 *	31.081024 *	31.115961 *
(41 , 10)	32.	31.964410 *	31.808928 *	31.696755 *	31.719443 *	31.628984 *	31.679548 *

129 (42 , 10)	24.	24.296115 *	24.531951 *	24.589184 *	24.422447 *	25.029404 *	25.005142
130 (55 , 10)	30.	30.447142 *	30.340159 *	30.451716 *	30.532539 *	30.458194 *	30.387919
131 (72 , 10)	35.	34.899185 *	34.316783 *	34.411277 *	34.399966 *	34.358180 *	34.516175
132 (89 , 10)	32.	31.529445 *	31.5365 4 *	31.866992 *	31.893144 *	31.917768 *	31.[illegible]
133 (120 , 10)	33.	30.969158 *	30.927841 *	31.249297 *	31.2.7970 *	31.[illegible] *	31.[illegible]
134 (126 , 10)	27.	25.878765 *	26.269632 *	26.436646 *	26.421230 *	26.[illegible] *	25.[illegible]
135 (158 , 10)	29.	29.464092 *	29.373866 *	29.[illegible] *	29.309365 *	29.377969 *	29.541894
136 (164 , 10)	30.	30.945586 *	30.938378 *	29.889997 *	29.882497 *	29.812586 *	29.801719
137 (258 , 10)	25.	26.357375 *	26.568979 *	26.125830 *	26.071387 *	26.044912 *	26.038519
138 (30 , 11)	88.	88.734890 *	88.210644 *	87.738821 *	87.6[illegible] *	87.635723 *	88.016467
139 (72 , 11)	92.	94.868020 *	92.167651 *	92.214533 *	92.260166 *	91.556433 *	92.594773
140 (73 , 11)	89.	91.364216 *	90.806868 *	90.871917 *	90.785321 *	91.316158 *	90.747504
141 (86 , 11)	81.	81.763514 *	81.882239 *	81.799416 *	81.890824 *	81.[illegible]962 *	80.563558
142 (111 , 11)	88.	9[illegible] *	89.552997 *	89.475842 *	89.507572 *	89.305121 *	89.482329
143 (117 , 11)	89.	89.031185 *	88.455567 *	88.466653 *	88.365716 *	88.724360 *	87.745534
144 (153 , 11)	79.	77.242029 *	79.129646 *	79.085172 *	79.182551 *	78.854538 *	78.623418
145 (149 , 11)	84.	86.826348 *	86.080304 *	86.244753 *	86.202489 *	86.312352 *	[illegible]
146 (152 , 11)	80.	79.83[illegible]81 *	80.146[illegible]57 *	80.022270 *	80.019645 *	80.258292 *	[illegible]
147 (157 , 11)	84.	84.938837 *	84.281814 *	84.261349 *	84.314044 *	83.696468 *	[illegible]
148 (170 , 11)	83.	86.376416 *	85.954159 *	85.974684 *	86.[illegible]1199 *	89.[illegible] *	[illegible]
149 (183 , 11)	94.	88.897779 *	88.[illegible]64237 *	88.233894 *	88.070362 *	89.764698 *	[illegible]
150 (201 , 11)	88.	86.376416 *	84.976922 *	84.921827 *	84.988706 *	89.[illegible]51 *	[illegible]
151 (202 , 11)	89.	88.512396 *	87.955869 *	87.806442 *	87.765589 *	97.957664 *	97.[illegible]
152 (216 , 11)	84.	84.[illegible]182 *	84.274813 *	84.331464 *	84.275155 *	84.704357 *	83.[illegible]
153 (229 , 11)	85.	87.991712 *	96.883760 *	86.575148 *	86.679126 *	86.470012 *	86.[illegible]
154 (236 , 11)	78.	80.685636 *	79.751506 *	79.745286 *	79.873853 *	79.255735 *	[illegible]
155 (1 , 12)	88.	93.877704 *	89.297854 *	88.830358 *	88.506872 *	86.592725 *	[illegible]
156 (23 , 12)	84.	84.227785 *	82.285856 *	83.[illegible]53994 *	83.[illegible]96736 *	87.872306 *	[illegible]
157 (32 , 12)	83.	84.253518 *	82.699216 *	83.049692 *	83.[illegible]40935 *	82.057112 *	[illegible]
158 (115 , 12)	79.	81.742653 *	82.519521 *	81.842121 *	81.595764 *	78.111213 *	[illegible]
159 (124 , 12)	77.	77.[illegible]16539 *	79.[illegible]09284 *	79.141450 *	79.291617 *	78.495834 *	[illegible]
160 (148 , 12)	85.	81.837199 *	84.[illegible]40033 *	83.984854 *	84.317148 *	82.[illegible]494 *	[illegible]
161 (151 , 12)	83.	87.677812 *	87.194922 *	86.570736 *	86.538992 *	86.329321 *	[illegible]
162 (181 , 12)	87.	83.271859 *	81.326060 *	80.[illegible]340[illegible]32 *	80.584867 *	81.861083 *	[illegible]
163 (247 , 12)	79.	86.493730 *	86.024631 *	86.391980 *	85.839240 *	85.[illegible]27 *	[illegible]
164 (10 , 13)	24.	22.681525 *	23.182781 *	23.[illegible]12650 *	23.340199 *	23.348087 *	[illegible]
165 (40 , 13)	21.	22.452139 *	22.652075 *	22.705543 *	23.[illegible]66779 *	23.152033 *	[illegible]
166 (47 , 13)	21.	21.639794 *	21.689902 *	21.718998 *	21.658416 *	21.322899 *	[illegible]
167 (65 , 13)	20.	21.068828 *	20.248412 *	20.274759 *	20.[illegible]09025 *	20.189276 *	[illegible]
168 (85 , 13)	22.	22.760472 *	23.864162 *	23.630584 *	23.855668 *	23.866806 *	[illegible]
169 (89 , 13)	25.	22.860448 *	23.438471 *	23.655572 *	23.[illegible]7278 *	23.[illegible] *	[illegible]
17 (102 , 13)	24.	22.902017 *	23.479250 *	23.124095 *	22.849715 *	22.842418 *	[illegible]
171 (120 , 13)	22.	22.460[illegible]623 *	22.782637 *	22.947147 *	22.683154 *	22.630737 *	[illegible]
172 (140 , 13)	23.	22.543727 *	22.671399 *	22.702885 *	22.463819 *	22.[illegible] *	[illegible]
173 (161 , 13)	23.	23.135005 *	23.524637 *	23.[illegible]90674 *	23.769451 *	23.721997 *	[illegible]
174 (163 , 13)	22.	22.081055 *	22.292328 *	22.459079 *	22.247775 *	22.277509 *	[illegible]
175 (165 , 13)	22.	22.181531 *	21.919803 *	21.839739 *	22.013376 *	21.989369 *	[illegible]
176 (177 , 13)	18.	19.837116 *	18.754369 *	18.451300 *	18.479[illegible]16 *	18.[illegible] *	[illegible]
177 (31 , 14)	78.	73.372293 *	73.[illegible]55 *	72.[illegible]668 *	72.133493 *	73.454419 *	[illegible]
178 (91 , 14)	73.	72.698579 *	73.689834 *	72.624020 *	73.594080 *	71.665643 *	[illegible]
179 (149 , 14)	80.	74.[illegible]57184 *	74.736594 *	76.062447 *	75.925227 *	75.838773 *	[illegible]
18 (160 , 14)	77.	77.745562 *	78.945629 *	77.516207 *	76.[illegible]0919 *	78.139798 *	[illegible]
181 (177 , 14)	62.	63.581062 *	62.[illegible]9[illegible]1 *	60.[illegible]63841 *	60.844043 *	63.[illegible] *	[illegible]
182 (182 , 14)	81.	78.665693 *	80.34390[illegible] *	79.613338 *	80.314325 *	78.746047 *	[illegible]
183 (190 , 14)	76.	71.[illegible]94076 *	71.437449 *	72.822864 *	74.501780 *	71.861879 *	[illegible]
184 (192 , 14)	77.	74.[illegible]19544 *	74.[illegible]77700 *	72.404545 *	71.245128 *	73.955726 *	[illegible]
185 (204 , 14)	55.	57.[illegible]74762 *	53.192935 *	54.511207 *	54.864843 *	56.[illegible]34 *	[illegible]
186 (207 , 14)	67.	66.[illegible]48208 *	64.568766 *	64.044919 *	62.826622 *	68.274069 *	[illegible]
187 (210 , 14)	70.	72.557958 *	72.556916 *	72.248194 *	75.623493 *	74.[illegible] *	[illegible]
188 (230 , 14)	73.	76.47[illegible]94 *	.77.[illegible]88185 *	75.[illegible]43977 *	75.459941 *	79.[illegible] *	[illegible]
189 (1 , 15)	66.	64.957992 *	63.824231 *	64.161385 *	64.953771 *	61.689759 *	[illegible]
19 (12 , 15)	60.	55.252228 *	55.571598 *	55.58[illegible]553 *	57.782132 *	57.[illegible] *	[illegible]
191 (14 , 15)	52.	55.[illegible]15870 *	55.264645 *	55.529863 *	53.554657 *	56.879999 *	[illegible]
192 (18 , 15)	59.	58.943448 *	58.464290 *	58.144939 *	60.425519 *	63.[illegible]193 *	[illegible]
193 (19 , 15)	50.	54.[illegible]87133 *	54.930444 *	54.721174 *	53.583810 *	53.761998 *	[illegible]
194 (37 , 15)	60.	61.020733 *	60.685544 *	60.[illegible]73790 *	63.321616 *	62.160305 *	[illegible]

195	(66 , 15)	52.	49.768056	50.752939	51.273715	[illegible]	[illegible]	[illegible]
196	(77 , 15)	57.	56.546113	56.757269	56.697216	57.126128	59.861635	58.15
197	(121 , 15)	57.	57.384113	57.451792	58.161775	[illegible]	58.091994	58.22
198	(123 , 15)	63.	59.875360	59.906516	[illegible]	63.655695	65.581833	[illegible]
199	(141 , 15)	60.	60.577581	[illegible]	61.638555	62.755606	[illegible]	62.03
200	(221 , 15)	56.	56.292024	56.141890	56.196018	54.091567	54.302306	53.69
201	(29 , 16)	65.	[illegible]	[illegible]	63.515586	63.710254	64.121570	64.25
202	(45 , 16)	62.	62.596693	63.169705	63.441431	63.585224	63.316622	[illegible]
203	(54 , 16)	57.	59.042324	58.814998	56.327333	58.001501	57.037490	[illegible]
204	(80 , 16)	60.	56.198795	57.680072	[illegible]	57.015754	56.747920	56.83
205	(104 , 16)	65.	66.269556	64.929846	[illegible]	65.276135	65.230908	65.29
206	(115 , 16)	63.	61.129394	61.574168	[illegible]	62.401094	63.553956	62.49
207	(143 , 16)	64.	63.760207	63.374246	[illegible]	63.918558	65.053770	65.42
208	(146 , 16)	55.	57.088553	57.118067	56.488961	56.092776	56.468959	56.55
209	(150 , 16)	62.	60.146113	[illegible]	[illegible]	61.730241	62.345738	62.35
210	(160 , 16)	69.	67.799742	[illegible]	65.220801	65.306427	65.759657	65.99
211	(171 , 16)	63.	62.703655	[illegible]	63.624953	63.732772	[illegible]	61.17
212	(203 , 16)	67.	65.346542	66.093992	66.326751	66.299524	66.293436	66.14
213	(215 , 16)	63.	[illegible]	[illegible]	[illegible]	61.817895	[illegible]	61.45
214	(231 , 16)	63.	63.363394	62.560059	61.615929	61.836606	62.529225	62.99
215	(265 , 16)	60.	62.675677	[illegible]	60.968254	61.337170	61.333575	61.25
216	(8 , 17)	60.	55.872293	57.182895	57.328847	57.901196	[illegible]	57.77
217	(12 , 17)	52.	51.030334	50.354684	50.323552	51.840300	51.969523	50.84
218	(15 , 17)	54.	53.120624	52.928007	52.350045	52.542103	[illegible]	52.24
219	(17 , 17)	54.	54.919792	[illegible]	56.716191	56.184980	56.141709	55.76
220	(20 , 17)	56.	52.161521	53.308740	53.422677	53.837213	[illegible]	54.03
221	(21 , 17)	55.	52.281701	52.907759	52.927927	52.667643	52.617337	53.09
222	(44 , 17)	52.	52.013246	51.399098	51.673347	51.395323	51.424113	51.60
223	(66 , 17)	43.	45.965214	43.754986	44.350121	44.110728	43.662969	43.75
224	(68 , 17)	47.	48.064800	45.778580	46.256901	45.497611	44.966421	45.11
225	(81 , 17)	53.	50.605196	48.969355	48.749277	48.634894	48.724915	48.47
226	(83 , 17)	61.	57.249127	58.713054	59.281468	59.113069	59.137133	59.17
227	(89 , 17)	56.	54.605176	55.475562	55.958858	55.936567	55.902223	55.78
228	(105 , 17)	62.	57.077313	58.920890	59.659883	59.246710	59.053454	59.20
229	(143 , 17)	56.	54.487523	55.336095	56.106923	56.299156	56.535014	56.26
230	(152 , 17)	50.	50.743818	50.413925	49.981211	50.138487	50.249143	50.05
231	(157 , 17)	56.	53.991052	54.749766	54.669826	54.554474	54.331632	54.5
232	(173 , 17)	52.	51.401426	51.543142	51.474300	50.878337	50.635650	[illegible]
233	(220 , 17)	50.	50.111736	49.399554	49.206409	49.250556	49.706722	49.7
234	(226 , 17)	57.	56.103673	57.756153	57.271065	58.219489	58.388217	58.4
235	(231 , 17)	55.	54.149425	54.972854	53.917077	54.320715	54.341168	54.2
236	(23 , 18)	128.	120.664366	124.248810	124.754906	124.103259	145.763390	148.5
237	(31 , 18)	123.	122.576388	[illegible]	121.469581	120.121075	119.590655	126.5
238	(35 , 18)	113.	119.236659	121.789996	121.892749	126.294564	119.440217	106.0
239	(100 , 18)	106.	112.326038	104.068908	103.548669	98.669630	101.717295	102.3
240	(109 , 18)	121.	[illegible]	118.657194	118.779095	121.394923	[illegible]	112.2
241	(115 , 18)	116.	117.104177	115.790279	115.289765	114.306510	97.618709	114.7
242	(118 , 18)	131.	128.910517	133.147507	133.158055	124.061977	135.959123	128.1
243	(130 , 18)	133.	[illegible]	132.605010	132.581476	130.674182	120.011348	117.2
244	(136 , 18)	136.	126.409630	132.355616	[illegible]	135.897706	133.810219	135.4
245	(138 , 18)	132.	123.553919	124.105603	123.058870	117.343172	[illegible]	108.7
246	(143 , 18)	134.	122.143953	[illegible]	126.146677	123.349717	107.711175	116.8
247	(162 , 18)	124.	119.704548	[illegible]	[illegible]	103.573152	103.277502	125.6
248	(171 , 18)	112.	[illegible]	[illegible]	117.859881	115.271689	127.636386	149.1
249	(174 , 18)	116.	114.681882	111.193857	111.190774	106.128743	104.507566	106.5
250	(195 , 18)	140.	128.338975	131.109019	130.323420	134.332988	145.991095	139.4
251	(220 , 18)	[illegible]	112.334809	109.839359	109.788093	105.296558	113.484591	115.7
252	(221 , 18)	113.	116.546808	118.217508	118.198713	123.619759	[illegible]	113.0
253	(222 , 18)	141.	112.483028	115.946755	[illegible]	116.896297	112.100781	89.4
254	(224 , 18)	126.	119.626053	122.816907	122.998713	115.898751	107.565845	111.8
255	(228 , 18)	115.	113.823943	[illegible]	106.202116	99.910031	94.205617	104.9
256	(236 , 18)	127.	[illegible]	120.244341	120.443878	123.146757	119.343643	101.6
257	(243 , 18)	123.	119.704548	116.656437	116.288883	117.560218	117.629693	102.2
258	(20 , 19)	153.	136.418612	139.840283	[illegible]	163.793635	169.822580	170.1
259	(30 , 19)	141.	[illegible]	149.749420	124.442330	117.357441	[illegible]	124.5
260	(42 , 19)	86.	[illegible]	97.258521	[illegible]	107.476458	84.016822	78.2

261	(43 , 19)	119.	121.185805	*	116.227622	*	118.101744	*	106.992987	*	105.225105	*	106.5207
262	(47 , 19)	138.	135.184085	*	135.352413	*	138.628506	*	135.423609	*	124.359749	*	116.0698
263	(51 , 19)	152.	137.253165	*	141.227569	*	177.949359	*	156.874975	*	162.488961	*	142.5359
264	(55 , 19)	133.	137.946749	*	141.310833	*	137.217155	*	133.569714	*	137.536214	*	142.3823
265	(78 , 19)	110.	119.329973	*	119.644691	*	152.383610	*	136.319887	*	107.856389	*	93.7292
266	(79 , 19)	119.	119.515557	*	113.312244	*	130.807212	*	113.758174	*	102.148807	*	89.7019
267	(111 , 19)	142.	149.975943	*	152.814151	*	148.567738	*	157.759962	*	158.547798	*	159.5557
268	(119 , 19)	128.	135.731335	*	135.349596	*	145.447754	*	157.923117	*	158.277933	*	160.2953
269	(129 , 19)	147.	146.981865	*	151.108584	*	126.928643	*	135.731626	*	147.284351	*	150.4291
270	(137 , 19)	144.	148.188155	*	153.084708	*	148.494726	*	144.767951	*	140.539366	*	141.5339
271	(198 , 19)	140.	135.295125	*	134.505177	*	117.391413	*	131.735614	*	124.147874	*	121.7189
272	(206 , 19)	130.	131.671254	*	128.393068	*	139.516864	*	140.860763	*	128.174898	*	131.1321
273	(210 , 19)	125.	141.419161	*	141.377787	*	144.923961	*	129.134241	*	128.831142	*	135.7219
274	(229 , 19)	137.	146.278540	*	151.178026	*	135.128732	*	144.646179	*	155.776713	*	153.9367
275	(9 , 20)	23.	25.377817	*	25.179745	*	25.068684	*	25.490919	*	25.349399	*	25.3978
276	(48 , 20)	24.	22.895579	*	24.275010	*	24.187695	*	24.146971	*	24.286232	*	24.2331
277	(67 , 20)	25.	23.793690	*	24.771713	*	24.659719	*	24.971583	*	25.310043	*	25.6674
278	(96 , 20)	24.	25.814706	*	25.174589	*	24.819125	*	24.772124	*	24.419600	*	23.9194
279	(124 , 20)	26.	23.933513	*	24.565232	*	24.402876	*	24.507805	*	24.684044	*	24.9058
280	(128 , 20)	26.	27.237704	*	26.782949	*	26.564573	*	26.367626	*	26.349643	*	26.5223
281	(136 , 20)	27.	27.427726	*	26.693997	*	26.636188	*	26.690752	*	26.852994	*	27.0177
282	(141 , 20)	26.	27.205971	*	26.678426	*	26.361222	*	25.952655	*	26.085221	*	26.0973
283	(146 , 20)	24.	23.723428	*	23.736610	*	23.887146	*	24.388794	*	24.450234	*	24.4243
284	(167 , 20)	26.	25.922338	*	25.686612	*	25.656802	*	25.321961	*	25.333963	*	25.4157
285	(169 , 20)	26.	25.874215	*	26.025020	*	25.957032	*	26.047208	*	26.301862	*	26.3067
286	(170 , 20)	26.	26.698346	*	26.465665	*	26.418726	*	26.475530	*	26.647382	*	26.6721
287	(171 , 20)	26.	26.456372	*	26.396145	*	26.352017	*	26.190756	*	26.156887	*	27.6961
288	(184 , 20)	21.	27.983125	*	27.076818	*	26.808711	*	26.624020	*	26.574494	*	26.0834
289	(201 , 20)	26.	26.698346	*	25.892822	*	26.032451	*	26.135243	*	25.944957	*	25.4027
290	(234 , 20)	27.	26.178995	*	26.012360	*	26.019725	*	26.687519	*	26.319015	*	27.0270
291	(240 , 20)	25.	24.430520	*	24.437118	*	24.738561	*	25.200429	*	25.089045	*	24.8699
292	(246 , 20)	27.	24.879670	*	25.258028	*	25.469691	*	25.431197	*	25.594538	*	25.5847

PR 50

Valeurs propres

		1	2	3	4	5
Original	λ	0,0010	0,00033	0,00016	0,00012	0,00011
	test 954	519	376	307	254	207
	%	45,58	14,98	7,21	5,63	4,87
45 colonnes otées	λ	0,0010	0,0034	0,00017	0,00012	0,00011
	test 804	438	317	257	214	173
	%	45,5	15,09	7,42	5,37	5,10
Trous	λ	0,002	0,0018	0,0012	0,0011	0,00083
	test 4890	4023	3268	2741	2253	1896
	%	17,73	15,43	10,79	9,97	7,29
Λ_1	λ	0,0010	0,00033	0,00016	0,00012	0,00011
	test 952	515	374	305	252	206
	%	45,89	14,79	7,25	5,62	4,83

Corrélation

45 colonnes otées - Orig.	$\mathcal{C}_I$	0,9996	0,9989	0,997	0,979	0,978
	σ_I	0,01	0,05	0,07	0,20	0,21
	$\mathcal{C}_J$	0,9997	0,996	0,996	0,978	0,978
	σ_J	0,02	0,10	0,09	0,21	0,22
Trou-Orig.	$\mathcal{C}_I$	0,30	0,30	0,001	0,12	0,05
	σ_I	1,49	2,91	3,03	3,21	3,96
	$\mathcal{C}_J$	0,30	0,43	0,07	0,08	0,22
	σ_J	1,09	2,75	3,46	2,74	2,73
Λ_1 - Orig.	$\mathcal{C}_I$	0,9998	0,995	0,998	0,9987	0,9988
	σ_I	0,02	0,10	0,06	0,05	0,05
	$\mathcal{C}_J$	0,99998	0,9998	0,9992	0,9998	0,9996
	σ_J	0,01	0,02	0,04	0,02	0,03

Valeurs propres

		1	2	3	4	5
Original	λ	0,0010	0,00033	0,00016	0,00012	0,0001
	test	519	376	307	254	207
	%	45,58	14,98	7,21	5,63	4,87
24 colonnes otées	λ	0,001	0,00033	0,00017	0,00012	0,0001
	test	476	346	280	231	189
	%	44,76	15,07	7,61	5,69	4,90
Trous	λ	0,0013	0,001	0,0007	0,0006	0,0004
	test	2224	1795	1477	1202	102C
	%	19,52	15,50	11,53	9,95	6,56
Λ_1	λ	0,0010	0,00033	0,00016	0,00012	0,0001
	test	517	375	306	253	207
	%	45,72	14,92	7,23	5,62	4,83

Corrélation

24 otées-Orig.	$\mathcal{C}_I$	0,99997	0,9994	0,99895	0,96	0,956
	ν_I	0,02	0,04	0,05	0,27	0,30
	$\mathcal{C}_J$	0,99975	0,997	0,997	0,96	0,96
	ν_J	0,03	0,08	0,08	0,28	0,28
Trous-Orig.	$\mathcal{C}_I$	0,88	0,45	0,07	0,9	0,17
	ν_I	0,53	2,43	2,32	2,55	2,11
	$\mathcal{C}_J$	0,97	0,58	0,07	0,19	0,17
	ν_J	0,27	2,45	3,17	1,82	2,11
Λ_1 - Orig.	$\mathcal{C}_I$	0,99991	0,998	0,998	0,9992	0,999
	ν_I	0,01	0,56	0,06	0,04	0,04
	$\mathcal{C}_J$	0,99999	0,99993	0,9992	0,9997	0,999
	ν_J	0,01	0,12	0,04	0,02	0,02

PR 100

Valeurs propres

		1	2	3	4	5
Original	λ	0,0010	0,00033	0,00016	0,00012	0,00011
	test	519	376	307	254	207
	%	45,58	14,98	7,21	5,63	4,87
83 colonnes otées	λ	0,0011	0,00037	0,00018	0,00013	0,0001
	test	381	271	217	179	146
	%	46,37	15,46	7,64	5,3	4,73
Trous	λ	0,0026	0,0024	0,0021	0,0017	0,0015
	test	7115	6088	5208	4475	3826
	%	13,35	12,51	10,72	8,93	7,90
Λ_1	test	0,0010	0,00032	0,00016	0,00012	0,0001
		508	370	302	249	204
	%	46,2	14,64	7,21	5,61	4,78

Corrélation

83 otées - Orig.	$\mathcal{C}_I$	0,99990	0,9990	0,994	0,959	0,948
	ν_I	0,03	0,05	0,11	0,29	0,32
	$\mathcal{C}_J$	0,9994	0,996	0,986	0,949	0,95
	ν_J	0,06	0,11	0,18	0,33	0,31
Trou - Orig.	$\mathcal{C}_I$	0,16	0,23	0,11	0,07	0,008
	ν_I	1,75	2,77	3,85	3,93	4,11
	$\mathcal{C}_I$	0,38	0,30	0,19	0,17	0,19
	ν_J	1,27	2,76	3,20	3,21	3,22
Λ_1 - Orig.	$\mathcal{C}_I$	0,9992	0,988	0,989	0,992	0,994
	ν_I	0,04	0,15	0,14	0,13	0,11
	$\mathcal{C}_J$	0,99997	0,9996	0,998	0,9986	0,998
	ν_J	0,01	0,03	0,06	0,06	0,06

Approximation de rang **1** pour les tableaux PR

i,j	p(i,j)	PR292	PR197	PR100	PR50	PR25
17,1	312	309,25	308,55			
56,1	308	311,68	311,53	311,44	311,45	
58,1	209	203,90	203,66			
64,1	318	322,61	322,48			
65,1	276	275,92	275,60			
67,1	265	263,25	263,01			
84,1	318	319,28	319,15			
89,1	317	308,6	309,2			
98,1	304	304,01	303,88			
143,1	321	308	312,95			
146,1	271	272,37	269,69	269,81	269,83	269,
164,1	297	300,08	300,05			
193,1	305	303,72	303,56			
238,1	315	306,5	306,34			
18,2	101	103,76	103,75	103,78		
135,2	112	108,15	108,15	108,11		
138,2	102	101,49	101,49			
143,2	109	102,01	104,45			
144,2	95	95,59	95,55	95,19	95,19	
159,2	100	98,63	98,59	98,59		
208,2	90	90,26	90,21	90,20		
228,2	85	87,79	87,75			
231,2	99	101,51	101,47	101,5		
87,3	45	45,86	45,85			
96,3	51	47,33	47,31			
126,3	40	43,67	43,75			
202,3	49	50,84	50,85	50,9	50,9	
212,3	48	48,16	48,15	48,18	48,18	48,
252,3	43	45,09	45,14	45,06	45,05	45,
29,4	156	152,78	152,48			
32,4	152	147,59	147,58	147,58	147,6	
38,4	151	148,15	148,14			
49,4	149	145,7	145,71	145,73	145,75	
76,4	125	125,78	125,69			
98,4	148	147,73	147,71			
171,4	150	147,90	147,47	146,91	146,94	146
191,4	125	129,12	129,04			
220,4	142	-	137,52	137,50	-	
237,4	134	136,01	135,18	135,16	135,18	
27,5	117	117,05	117,06			
72,5	123	122,52	122,78			
96,5	117	110,85	110,82			
142,5	118	119,11	119,11			
165,5	115	114,36	114,4			
183,5	121	119,18	119,4	119,49	119,49	
39,6	46	51,4	51,41			
95,6	49	48,97	49			
114,6	48	49,25	49,28	49,26		
223,6	47	47,16	47,21			
226,6	46	47,85	47,9			
69,7	48	47,22	47,19	47,26		
97,7	48	48,92	48,95	48,95	48,95	
107,7	48	46,12	46,10			
109,7	48	47,10	47,10			
144,7	49	45,59	45,66			

156,7	47	45,82	45,77	45,86	45,88	45,87
158,7	43	44,36	44,34	44,37		
194,7	44	46,19	46,20			
233,7	48	47,86	47,84			
85,8	38	39,52	39,54	39,49		
105,8	40	40,71	40,7			
119,8	41	41,97	41,98	41,99	42,38	
163,8	44	41,59	41,62			
173,8	40	40,66	41,08			
181,8	41	40,52	40,49			
184,8	37	42,04	42,02			
187,8	44	40,94	40,95			
210,8	45	43,44	43,45	43,47		
9,9	43	47,05	47,06	47,06		
25,9	47	42,90	42,88	42,87	42,84	42,84
28,9	50	49,25	49,25	49,25		
40,9	50	48,36	48,32			
67,9	42	43,88	43,84			
68,9	42	43,3	43,26			
73,9	52	52,34	52,36	52,31	52,28	52,29
107,9	51	50,44	50,47	50,54	50,53	
131,9	47	47,46	47,48	47,48	47,46	47,47
141,9	53	50,41	50,42			
154,9	42	45,11	45,08	45,07		
183,9	50	50,81	50,96	50,96	50,94	50,95
187,9	49	48,54	48,56	48,66	48,64	48,64
227,9	45	44,63	44,6	44,58		
7,10	36	32,39	32,39	32,39		
12,10	30	29,54	29,66			
36,10	32	31,37	31,33			
42,10	24	24,53	24,56			
55,10	30	30,34	30,29			
72,10	35	34,32	34,37			
89,10	32	31,54	31,57	31,71	31,74	
120,10	33	30,93	30,92	30,93		
164,10	30	30,94	30,97	30,92	30,94	30,94
258,10	25	26,57	26,58	26,57	26,58	
30,11	88	88,21	88,23			
72,11	92	92,16	92,39			
73,11	89	90,81	90,83			
111,11	88	89,55	89,58	89,56	89,55	89,54
117,11	89	88,46	88,48	88,46	88,46	
133,11	79	79,13	79,12	79,14	79,14	79,15
157,11	84	84,28	84,31	84,33	84,33	
170,11	83	85,95	85,94	85,92	85,92	85,9
201,11	88	84,98	85,01			
202,11	89	87,96	87,98			
216,11	84	84,27	84,29			
229,11	85	86,88	86,94			
236,11	78	79,75	80,10	80,08	80,24	
1,12	88	89,3	89,43			
23,12	84	82,29	82,18			
32,12	83	82,7	82,68	82,66	82,65	
115,12	79	82,92	82,48	82,52		

181,12	87	81,33	81,28			
10,13	24	23,18	23,17			
85,13	22	23,86	23,86	23,86		
120,13	22	22,78	22,77	22,76	22,78	
140,13	23	22,67	22,65	22,65		
161,13	23	23,52	23,53	23,52		
165,13	22	21,92	21,81	21,81	21,81	21,80
31,14	78	73,13	73,18	73,18		
91,14	73	73,69	73,69			
160,14	77	78,95	78,99	78,96	79	
182,14	81	80,34	80,34			
190,14	76	71,44	71,50	71,47		
204,14	55	53,19	53,22	53,30	53,3	53,32
210,14	70	72,56	72,56	72,54		
14,15	52	55,36	55,37	55,38		
18,15	59	58,46	58,47			
37,15	60	60,69	60,70	60,71		
66,15	52	50,75	50,75	50,74		
77,15	57	56,76	56,76			
121,15	57	57,45	57,45	57,47	57,46	
123,15	63	59,91	60,07			
221,15	56	56,14	56,10			
29,16	65	63,93	63,72	63,86	63,88	63,87
45,16	62	63,11	63,11	63,10		
54,16	57	58,81	58,83	58,83	58,85	
80,16	60	57,68	57,68	57,67		
104,16	65	64,93	64,95	64,95	64,97	64,96
115,16	63	61,57	61,57	61,58		
143,16	64	63,37	63,39	63,56		
171,16	63	63,48	63,09	63,07	63,10	
215,16	63	61,17	61,17	61,16		
231,16	63	62,56	62,58	62,56		
265,16	60	61,51	61,52			
8,17	60	57,18	57,19	57,25	57,25	
15,17	54	52,93	52,91			
21,17	55	52,91	52,91	52,95		
44,17	52	51,4	51,39	51,40	51,41	51,45
66,17	43	43,75	43,73			
21,17	47	45,78	45,75			
44,17	53	48,97	48,95	48,94	48,95	
83,17	61	58,71	58,72			
89,17	56	55,48	55,62			
105,17	62	58,92	58,94			
157,17	56	54,75	54,75			
173,17	52	51,54	51,51			
220,17	50	49,4	49,25	49,27		
226,17	57	57,76	57,50	57,61		
231,17	55	54,97	54,97	55,02		
35,18	113	121,79	121,93			
100,18	106	104,07	104,36	104,45		
109,18	121	118,66	118,85			
130,18	133	132,61	132,80			
136,18	136	132,36	132,53			

138,18	132	124,11	124,32	122,45		
162,18	124	118,73	118,96	118,97	119,06	119,0
171,18	112	118,02	118,82			
174,18	116	111,19	111,43	111,49		
220,18	109	109,84	109,72	109,78		
221,18	113	118,22	118,43	118,40		
224,18	126	122,82	122,99			
228,18	115	107,05	107,28			
236,18	127	120,24	120	119,95		
243,18	123	116,66	116,91			
30,19	141	149,75	149,69			
42,19	86	97,26	97,47			
47,19	138	135,35	135,49			
51,19	152	141,23	141,11			
55,19	133	141,31	141,38	141,45		
111,19	142	152,81	152,74	152,77	152,62	152,5
119,19	128	135,35	135,33	135,24		
137,19	144	153,08	152,98			
198,19	140	134,51	134,52			
206,19	130	128,39	128,43			
210,19	125	141,38	141,35	141,26		
229,19	137	151,18	151,13	151,16	151,01	
9,20	23	25,18	25,18	25,15		
67,20	25	24,77	24,78	24,79	24,8	24,8
96,20	24	25,17	25,17	25,42	25,41	25,4
124,20	26	24,57	24,49	24,5	24,5	24,5
128,20	26	26,78	26,78	26,74		
141,20	26	26,68	26,68			
146,20	24	23,74	23,67	23,65	23,65	
167,20	26	25,68	25,68	25,65		
169,20	26	26,03	26,03			
184,20	21	27,08	27,07			
201,20	26	25,89	25,90	25,9		
246,20	27	25,26	25,27	25,25		

Le point (220,4) s'est trouvé changé en

(230,4)	154	156,42	dans PR292
(223,4)	142	142,94	PR50

Valeurs propres

PR 197 : 3,7%de trous

	1	2	3	4	5
λ_1 test %	0,001 497 46,60	0,001 515 45,81	0,001 524 45,45	0,001 526 45,66	0,001 538 44,99
λ_2 test %		0,00036 360 16,33	0,00035 372 15,86	0,00035 373 15,81	0,00035 386 15,56
λ_3 test %			0,00018 292 8,27	0,00018 295 8,06	0,0002 298 8,94
λ_4 test %				0,00013 240 5,67	0,00013 244 5,58
λ_5 test %					0,00011 196 4,87

PR192 : 5,5% de trous

	1	2	3	4	5
λ_1 test %	0,001 481 47,49	0,001 519 45,31	0,001 514 45,77	0,001 525 46,01	0,001 540 45,65
λ_2 test %		0,0004 344 18,41	0,00037 352 17,08	0,00036 368 16,17	0,00037 379 16,22
λ_3 test %			0,00015 285 7,04	0,00018 290 8,07	0,0002 294 8,56
λ_4 test %				0,00013 234 5,72	0,00013 239 5,48
λ_5 test %					0,00011 192 4,81

De tous ces résultats, il convient de noter que les nombres calculés sont très proches qu'il s'agisse d'approximations d'ordre différent (0, ...5) pour un même nombre de trous ou qu'il s'agisse de nombres de trous différents, 25,50. ou même 292 trous, soit de 0,2% à 5,5% de trous.

Quant aux analyses - originales, avec des colonnes ôtées, avec les nombres calculés -, elles sont toutes très voisines puisque les coefficients de corrélati entre les valeurs des facteurs pour I ou J varient de 0,95 à 1.

De plus, on remarque que le nombre d'itérations nécessaires augmentent avec le nombre de trous ainsi qu'avec le rang de l'approximation.

PR	25	50	100	197	292
nb it.	7	7	8	10	16
CI	9	23	55		
nb it.	8	14	19		

VIII - CONCLUSION

L'étude précédente montre que ce programme donne un outil valable d'étude de données en cas de données manquantes ou même de données erronées.

Cet outil est d'autant plus sûr (ex. PR et MU) que les différentes approximations d'ordre successif 1...5 donnent des résultats très proches. Il est aussi très intéressant de remarquer que ces résultats ne dépendent que très peu du nombre de trous, seul le nombre d'itérations et par conséquent le temps en dépendant très directement.

Seulement, cette méthode n'a été utilisée, comme je l'ai déjà indiqué, que su des données pour lesquelles la première valeur propre est grande par rapport aux autres. Il serait donc intéressant de vérifier si, même dans un autre cas l'approximation d'ordre 1 reste suffisante comme dans tous les exemples.

Il serait aussi intéressant d'étudier cette méthode lorsque le tableau de dor nées est non plus composé de nombres pratiquement tous différents de 0, mais d'un très grand nombre de zéros, par exemple les données sur les départements français ou celles sur les phrases espagnoles.

De plus, ainsi qu'on l'a vu pour les praomys, lorsqu'on se voit obligé de supprimer des lignes ou des colonnes, faute de données exactes, il paraît être intéressant de reprendre une étude (ex. celle des praomys) où on a été amené à supprimer un certain nombre de colonnes ou lignes et d'effectuer l'analyse du tableau après l'avoir complété par cette méthode.

A N N E X E S

1 - Approximation de rang 0

Les temps de calcul de ce programme sont très faibles. Pour un passage sur l'ordinateur CII* de la faculté des Sciences, cela a pris :

1e + 2e méthode à la suite		1e méthode seule	
MU	1,087 mn	MU + CI184	1,143 mn
CI9 + CI23 + CI55 +CA91	1,834 mn		
CI105 + CA191	2,01 mn		
CA11 + CA41	1,834 mn		

*un CII 10070

```
      DIMENSION P(100,100),IT(200),JT(200),PI(100),P11(100),A(200)
      DIMENSION PJ(100),P1J(100),A1(200)
      READ (5,30) MMAX
      READ (5,18) EPS
      READ (5,10) IMAX,JMAX,NMAX,KIT
      DO 100 I=1,IMAX
  100 READ (5,11) (P(I,J),J=1,JMAX)
      READ (5,10) (IT(N),N=1,NMAX)
      READ (5,10) (JT(N),N=1,NMAX)
   10 FORMAT (4OI2)
   11 FORMAT (2OF4.2)
      WRITE (6,12) IMAX,JMAX,NMAX,KIT,EPS
   12 FORMAT (3H I=,I4,5H , J=,I4,5H , N=,I4,6H ,KIT=,I4,6H ,EPS=,F8.5)
      JLIG=20
      JF=(JMAX-1)/JLIG+1
      DO 101 JJ=1,JF
      JDEP =(JJ-1)*JLIG+1
      JFIN = JJ*JLIG
      IF(JFIN.GT.JMAX) JFIN=JMAX
      WRITE (6,13)  (J,J=JDEP,JFIN)
      WRITE (6,14)
      DO 101 I=1,IMAX
  101 WRITE (6,15) I,(P(I,J),J=JDEP,JFIN)
   13 FORMAT (1HO/1HO,8X,1H*,20(I4,2X))
   14 FORMAT (1H ,64(2H**))
   15 FORMAT (1H ,I4,3X,2H* ,20F6.2)
      DO 102 N=1,NMAX
  102 WRITE(6,16) N,IT(N),JT(N)
   16 FORMAT (I5,10H         ( ,I4,2H ,,I4,2H ))
      DO 203 N=1,NMAX
      I=IT(N)
      J=JT(N)
      A(N) = P(I,J)
  203 P(I,J) =0.
      DO 103 JJ=1,JF
      JDEP =(JJ-1)*JLIG+1
      JFIN = JJ*JLIG
      IF(JFIN.GT.JMAX) JFIN=JMAX
      WRITE (6,13)  (J,J=JDEP,JFIN)
      WRITE (6,14)
      DO 103 I=1,IMAX
  103 WRITE (6,15) I,(P(I,J),J=JDEP,JFIN)
      DO 104 I=1,IMAX
  104 PI(I) = 0.
      PT = 0.
      DO 105 J=1,JMAX
      PJ(J) = 0.
      DO 105 I=1,IMAX
      PIJ = P(I,J)
```

```
      PJ(J) = PJ(J) + PIJ
      PI(I) = PI(I) + PIJ
  105 P1 = PT + PIJ
      P1 = PT
      DO 106 I=1,IMAX
  106 PII(I) = PI(I)
      DO 107 J=1,JMAX
  107 P1J(J) =PJ(J)
      WRITE (6,17)
   17 FORMAT ( 1H1,30H   PREMIERE  METHODE            )
      GO
  110 K = K + 1
      IF (P.GT.PT) GO TO 111
      DO 108 N=1,NMAX
      I = IT(N)
      J = JT(N)
      A1(N) = P(I,J)
      PIJ = PII(I)*P1J(J)/P1
      PII(I) = PII(I) + PIJ - A1(N)
      P1J(J) = P1J(J) + PIJ - A1(N)
      P1 = P1 + PIJ - A1(N)
  108 P(I,J) = PIJ
      DO 109 N=1,NMAX
      I = IT(N)
      J=JT(N)
      PIJ = P(I,J)
      IF (ABS(A1(N)-PIJ).GT.EPS) GO TO 110
  109 CONTINUE
  111 WRITE (6,28)  K
   28 FORMAT (2CH NB D'ITERATIONS =   ,I4)
      WRITE(6,19)
   19 FORMAT (1HC/30H  VALEUR  REELLE  CALCULEE    )
      DO 112 N= 1,NMAX
      I=IT(N)
      J = JT(N)
  112 WRITE (6,20) N,I,J,A(N),P(I,J)
   20 FORMAT (1H ,I4,2H (,I3,2H ,,I3,2H ), F9.5,F9.5)
      DO 113 JC=1,JF
      JDEP =(JC-1)*JLIG+1
      JFIN = JC*JLIG
      IF(JFIN.GT.JMAX) JFIN=JMAX
      WRITE (6,13)  (J,J=JDEP,JFIN)
      WRITE (6,14)
      DO 113 I=1,IMAX
  113 WRITE (6,15) I,(P(I,J),J=JDEP,JFIN)
      WRITE (6,31)
   31 FORMAT (1H1)
  400 CONTINUE
      END
```

L'algorithme 2 est obtenu en changeant sur ce programme la ligne 70 et en la remplaçant par :

CIJ = A1(N)

PIJ = (P1I(I)-CIJ) x (P1J(J)-CIJ) / (P1 + CIJ - P1I (I) - P1J(J))

Pour ce programme, les dimensions sont définies ainsi :

P	dimension	card I x card J
IT,JT,A,A1	dimension	N = card $\mathcal{N}$, nombre de trous
PJ, P1J	dimension	card J
PI, P1J	dimension	card I

Ce programme fait les calculs point par point.

Pour les avoir par tour comme sont expliqués les algorithmes, il suffit de remplacer ainsi les lignes 71 à 74:

```
108 P(I,J) = PIJ
    DØ 200 N = 1,NMAX
    I = IT(N)
    J = JT(N)
    PIJ = P(I,J)
    P1I(I) = P1I(I) + PIJ - A1(N)
    P1J(J) = P1J(J) + PIJ - A1(N)
    P1 = P1 + PIJ - A1(N)
200 CØNTINUE
```

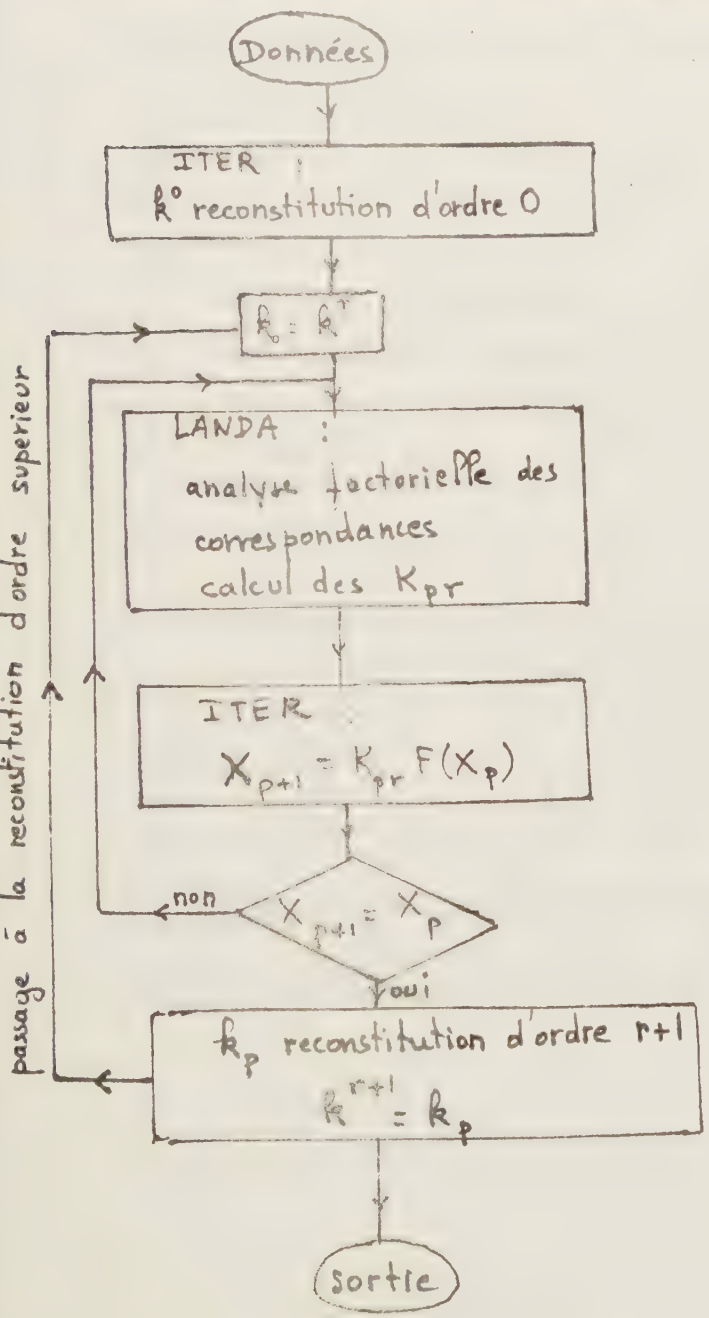
2 - Approximation de rang r

Ce programme passé sur l'I.B.M. 370 - 165 du CIRCE a pris pour obtenir les 6 approximations recherchées (r = 0, ...5).

MU CI 9, 23,55 CA 11, 41, 91 PR 100	à la suite 18,39 mn (les 7 premières, moins de 6mn)
PR 25,50	27,02mn
PR 197,292	15,49 mn

NOTICE D'UTILISATION DU PROGRAMME TROU

Ce programme écrit en FORTRAN contient un programme principal et trois sous-programmes appelés dans l'ordre ITER, LANDA, DIAG. Le diagramme joint schématise le déroulement du programme TROU.



k^r reconstitution d'ordre r

k_p tableau de travail constitué du tableau original complété par des valeurs provisoires X_p

X_p tableau des valeurs provisoires des "trous"

K_{pr} tableau des nombres $(1 + \sum_{i=1}^{r} \sqrt{\lambda_i}\, \varphi_{ip}\, \varphi_{iq})$ utilisé dans l'algorithme ITER pour rechercher les solutions du système $x = KF(x)$

LANDA, pour l'analyse factorielle des correspondances, fait appel au sous programme DIAG de diagonalisation d'une matrice carrée symétrique.

Ce sous programme DIAG est basé sur la méthode des puissances itérées. Il serait donc possible de le remplacer par une méthode pl rapide mais l'amélioration de la rapidité n'est pas évidente car or connaît toujours (sauf la première fois) une valeur approchée du vecteur propre recherché, ce qui diminue particulièrement le nombre d'itérations nécessaires à l'intérieur même de ce sous programme D.

Ce programme trou utilise les notations suivantes :

MMAX nombre d'analyses

IMAX nombre d'individus (card I) : card I $\geqslant$ card J

JMAX nombre de variables (card J)

NMAX nombre de trous (N)

P tableau des données de l'analyse considérée

IT tableau des indices I des trous

JT tableau des indices J des trous

KIT nombre maximum d'itérations admis lors de la recherche des solutions du système X = KF(X) dans l'algorithme ITER

NBIT nombre maximum de valeurs propres (noté NB dans cette notice) On obtiendra alors les reconstitutions d'ordre 0,1, .. NBIT

LIT nombre d'itérations maximum de l'algorithme trou pour chaque reconstitution recherchée d'ordre r ($r \geqslant 1$). La reconstitutic d'ordre 0 ne faisant intervenir que la valeur propre triviale est obtenue par l'algorithme ITER seul

EPS précision demandée pour chaque reconstitution(dans les calcul effectués, elle a été prise égale à 10^{-4} en général)

FMT format variable d'entrée des données.

UTILISATION PRATIQUE DU PROGRAMME TROU

Pour utiliser le programme trou, il convient d'adapter les dimensions des tableaux et d'entrer les données.

1°- Dimensions des tableaux

Programme principal

Les tableaux doivent avoir au moins les dimensions suivantes :

Tableaux	Dimension minimale
IT, JT	nombre de trous N
B	card J× (card J + 1)/2
PI	card I
PJ	card J
P	(card I, card J)
A2, A3, CØR, A	N
ATEST, PØURC	nombre de valeurs propres maximum NB
VAL	NB + 1
W	(NB + 1)× card J
AC	(NB + 1, N)

La dimension du tableau FMT permettant de lire le format variable res de 20, ce qui permet de lire un format tenant sur une carte perforée (80 colonnes de cette carte perforée).

Sous programme ITER

Tableau	Dimension minimale
IT,IJ	nombre de trous N
B	card J x (card J + 1)/2
P1J	card J
P1I	card J
P	(card I, card J)
A1, CØR	N

Les tableaux P1I, P1J, P, CØR, B doivent avoir exactement et respectivement les mêmes dimensions que les tableaux PI, PJ, P, CØR, B du programme principal.

Sous-programme LANDA

De la même façon, les tableaux de même nom dans LANDA et le programme principal ainsi que P1I et PI, P1J et PJ doivent avoir les mêmes dimensions.

Tableau	Dimension minimale
[illegible]	N
[illegible]	card J x (card J +1)/2
[illegible]	card J
[illegible]	card I
[illegible]	(card I, card J)
CØR	N
[illegible]	NB
[illegible]	NB + 1
G	(NB + 1) x card I
F, W	(NB + 1) x card J

Sous-Programme DIAG

[illegible]	card J NB + 1 (NB + 1) x card J card J x (card J + 1)/2

2°- Données à entrer

Ces données comportent

- une carte indiquant le nombre total d'analyses à effectuer à la suite. Ce nombre MMAX est lu en format 20I4.
- pour chaque analyse l'ordre des cartes doit être le suivant :

1- précision souhaitée EPS : format F8.6
(dans les études effectuées 10^{-4} a semblé largement suffisant)

2- IMAX, JMAX, NMAX, KIT : format 5I4
(dans les études effectuées KIT = 1000 a été rarement atteint)

3- FMT format variable des données : format 20A4

4- Le tableau des données considéré
lu selon le format variable FMT
le tableau est lu ligne par ligne, chaque ligne représentant les valeurs de p(i,j) pour i donné, j variant.

5- les indices I des trous format 25I3

6- les indices J des trous format 25I3

7- NBIT, LIT

Dans les études effectuées, NBIT a été pris égal à 5 (en fait seules des reconstitutions d'ordre ≤ 2 ont été retenues) et L égal à 100.

Il faut remarquer que les noms des individus et des variables n'interviennent pas dans ce programme et ne sont donc jamais lus.

Chaque trou est défini par la valeur de ces indices i et j.

3°- Sorties des résultats

Pour chaque analyse, ce programme écrit le tableau des données considérées, les reconstitutions trouvées d'ordre 0, 1, ... NBIT avec les valeurs propres finales pour chaque reconstitution d'ordre > 0.

Un tableau récapitulatif donne pour chaque trou les valeurs trouvées pour chaque reconstitution.

Le nombre d'itérations (L) écrit est le nombre d'itérations de l'algorithme trou complet ($L \leq LIT$). Pour le sous programme ITER, seul le dépassement du nombre d'itérations maximum KIT est indiqué.

Tel que le listage est donné, les valeurs des trous sont perforées à la fin de ce programme, ceci pour permettre l'introduction de ces valeurs dans le tableau original en vue d'une analyse factorielle des correspondances du tableau complet.

```
RAN IV G LEVEL  21                    MAIN                    DATE = 73211  20/23/46

1               COMMON/D/ IT(3[illegible]),JT(3[illegible]),B(21[illegible])
2               COMMON/E/ EPS,KIT,IMAX,JMAX,NMAX ,NBIT
3               DIMENSION PI(265),PJ(20 ),P(265,2[illegible])
4               DIMENSION A2(3[illegible]),A3(3[illegible]),COR(300),A(3[illegible])
5               DIMENSION FMT(20),ATEST(5),POURC(5),VAL(6),W(12[illegible])
6               DIMENSION AC(6,300)
7               DOUBLE PRECISION AC
8               DOUBLE PRECISION P,PI,PJ,PT,PIJ,COR,W,A2,A3,VAL,EPS,TEST
9               DOUBLE PRECISION ATEST,POURC    ,A
0               DOUBLE PRECISION B
1               READ (5,30) NMAX
2               DO 4[illegible] M=1,MMAX
3              30 FORMAT (5I4)
4              18 FORMAT (F8.6)
5               READ (5,18) EPS
6               READ (5,30) IMAX,JMAX,NMAX,KIT
7               WRITE (6,12) IMAX,JMAX,NMAX,KIT,EPS
8               READ(5,4 ) (FMT(I),I=1,2[illegible])
9              40 FORMAT (20A4)
0               DO 10[illegible] I=1,IMAX
1             100 READ (5,FMT) (P(I,J),J=1,JMAX)
2               READ (5,10) (IT(N),N=1,NMAX)
3               READ (5,10) (JT(N),N=1,NMAX)
4              10 FORMAT (25I3)
5              12 FORMAT (3H I#,I4,5H , J#,I4,5H , N#,I4,6H ,KIT#,I4,6H ,EPS#,F8
6               JLIG=2[illegible]
7               JF=(JMAX-1)/JLIG+1
8               DO 101 JJ=1,JF
9               JDEP =(JJ-1)*JLIG+1
0               JFIN = JJ*JLIG
1               IF(JFIN.GT.JMAX) JFIN=JMAX
2               WRITE (6,13)  (J,J=JDEP,JFIN)
3               WRITE (6,14)
4               DO 101 I=1,IMAX
5             101 WRITE (6,15) I,(P(I,J),J=JDEP,JFIN)
6              13 FORMAT (1H /1H[illegible],8X,1H*,2[illegible](I4,2X))
7              14 FORMAT (1H ,64(2H#*))
8              15 FORMAT (1H ,I4,3X,2H#[illegible],20F6.2)
9               DO 1[illegible]2 N=1,NMAX
0               I=IT(N)
1               J=JT(N)
2               A(N)=P(I,J)
3               P(I,J)=0.D00
4             102 WRITE(6,16) N,I,J,A(N)
5              16 FORMAT (I5,1[illegible]H          ( ,I4,2H ,,I4,2H ),4H     ,F6.0)
6               DO 1[illegible]4 I=1,IMAX
7             104 PI(I) = 0.D00
8               PT = 0.D00
9               DO 1[illegible]5 J=1,JMAX
0               PJ(J) = 0.D00
1               DO 105 I=1,IMAX
2               PIJ = P(I,J)
3               PJ(J) = PJ(J) + PIJ
4               PI(I) = PI(I) + PIJ
5             105 PT = PT + PIJ
6               DO 3[illegible]1 N=1,NMAX
7             301 COR(N) =1.D00
8               CALL ITER(P,PI,PJ,PT,COR)
```

```
0059                WRITE(6,19)
0060             19 FORMAT (1H /30H        VALEUR CALCULEE        )
0061                DO 112 N= 1,NMAX
0062                I=IT(N)
0063                J = JT(N)
0064                WRITE(6,20) N,I,J,A(N),P(I,J)
0065                AC(1,N) = P(I,J)
0066            112 CONTINUE
0067             20 FORMAT (1H ,I4,2H (,I3,2H ,,I3,2H ),F6.0,3X,F10.6)
0068                WRITE (6,31)
0069                READ (5,30) NBIT,LIT
0070                KM=(NBIT+1)*JMAX
0071                DO 302 K = 1,KM
0072            302 W(K)=0.D00
0073                KK=NBIT+1
0074                DO 700 K=1,KK
0075            700 W(K)=1.D00
0076                DO 310 NB =1,NBIT
0077                DO 311 N = 1,NMAX
0078                I = IT(N)
0079                J = JT(N)
0080            311 A2(N) = P(I,J)
0081                DO 320 L = 1,LIT
0082                DO 312 N = 1,NMAX
0083                I = IT(N)
0084                J = JT(N)
0085            312 A3(N) = P(I,J)
0086                CALL LANDA (P,PT,PI,PJ,COR,NB,W,VAL,TEST,ATEST,POURC)
0087                CALL ITER (P,PI,PJ,PT,COR)
0088                DO 322 N=1,NMAX
0089                I = IT(N)
0090                J = JT(N)
0091                IF(DABS(P(I,J)-A3(N)).GT.EPS) GO TO 320
0092                K=N
0093            322 CONTINUE
0094                IF(K.EQ.NMAX) GO TO 422
0095            320 CONTINUE
0096            422 WRITE (6,19)
0097                NB=NB+1
0098                WRITE (6,71) (II,II=1,NB)
0099                WRITE(6,72)
0100                WRITE (6,73) (VAL(N),N=2,NB)
0101                WRITE(6,74) TEST,(ATEST(N),N=1,NB)
0102                WRITE(6,75) (POURC(N),N=1,NB)
0103             71 FORMAT (1H ,7X,1HM,5X,5(1H*,7X,I3,9X))
0104             72 FORMAT (1H ,13(1H*),50(2H**))
0105             73 FORMAT (1H ,14H VALEUR PROPRE,5(1H*,E18.8,1X))
0106             74 FORMAT (1H ,4HTEST,E9.3,5(1H*,E18.8,1X))
0107             75 FORMAT (1H ,13H POURCENTAGE  ,5(1H*,F12.2,7X))
0108                DO 323 N = 1,NMAX
0109                I = IT(N)
0110                J = JT(N)
0111                WRITE(6,20) N,I,J,A(N),P(I,J)
0112                MBB = NB +1
0113                AC(MBB,N) = P(I,J)
0114            323 CONTINUE
0115                WRITE (6,5 ) L,NB
0116             5  FORMAT (1H1,12H NB D'ITER L,I4,12H  NB DE VAL ,I4)
```

```
FTRAN IV G LEVEL  21                    MAIN                    DATE = 73320   03/23/3
0117                IF(DABS(VAL(NB+1)).GT.1.D-5) GO TO 330
0118                WRITE (6,53)
0119             53 FORMAT (20H  VALEUR NULLE         )
0120                GO TO 400
0121            330 DO 331 N = 1,NMAX
0122                I = IT(N)
0123                J = JT(N)
0124                IF(DABS(P(I,J)-A2(N)).GT.EPS) GO TO 310
0125                K=N
0126            331 CONTINUE
0127                IF(K.EQ.NMAX) GO TO 410
0128            310 CONTINUE
0129                GO TO 340
0130            410 WRITE (6,55)
0131             55 FORMAT (20H  VALEUR STABILISEE )
0132            340 WRITE (6,56)
0133             56 FORMAT ( 36H  VALEUR REELLE CALCULEE FINALE       )
0134                WRITE(6,31)
0135             31 FORMAT (1H1)
0136                MBI = NBIT + 1
0137                DO 510 NB =1,MBI
0138                PUNCH511, (AC(NB,N),N=1,NMAX)
0139            510 CONTINUE
0140                DO 512 N=1,NMAX
0141                I=IT(N)
0142                J=JT(N)
0143                WRITE(6,513) N,I,J,A(N),(AC(NB,N),NB=1,MBI)
0144            512 CONTINUE
0145            513 FORMAT(1H ,I4,2H (,I3,2H ,,I3,2H ),F6.0,3X,6(F10.6,1X,2H #,2
0146            511 FORMAT(8F10.5)
0147            400 CONTINUE
0148                STOP
0149                END
```

```
      SUBROUTINE ITER (P,P1I,P1J,P1,COR)
      COMMON/D/ IT(300),JT(300),B(210)
      COMMON/E/ EPS,KIT,IMAX,JMAX,NMAX ,NBIT
      DIMENSION P1J(20 ),P1I(265),P(265,20 )
      DIMENSION A1(300),COR(300)
      DOUBLE PRECISION P,P1I,P1J,P1,COR,A1,PIJ,EPS
      DOUBLE PRECISION B
      K=0
  11  K = K + 1
      IF (K.GT.KIT) GO TO 111
      DO 108 N=1,NMAX
      I = IT(N)
      J = JT(N)
      A1(N) = P(I,J)
      PIJ = P1I(I)*P1J(J)*COR(N)/P1
      IF (PIJ.LE.0.D00) PIJ = 0.D00
 108  P(I,J) = PIJ
      DO 200 N=1,NMAX
      I = IT(N)
      J = JT(N)
      PIJ = P(I,J)
      P1I(I) = P1I(I) + PIJ - A1(N)
      P1J(J) = P1J(J) + PIJ - A1(N)
      P1 = P1 + PIJ - A1(N)
 200  CONTINUE
      DO 109 N=1,NMAX
      I = IT(N)
      J=JT(N)
      PIJ = P(I,J)
      IF(DABS(A1(N)-PIJ).GT.EPS) GO TO 110
 109  CONTINUE
 111  CONTINUE
      RETURN
      END
```

```
      SUBROUTINE LANDA(P,P1,P1I,P1J,COR,NB,W,VAL,TEST,ATEST,POURC
      COMMON/D/ IT(300),JT(300),B(210)
      COMMON/E/ EPS,KIT,IMAX,JMAX,NMAX ,NBIT
      DIMENSION P1J(20 ),P1I(265),P(265,20 ),COR(300)
      DIMENSION ATEST(5),POURC(5),VAL(6)
      DIMENSION G(1590),F(120),W(120)
      DOUBLE PRECISION B,P,P1I,P1J,P1,TEST,VAL,VAL2,F,W,G,ATEST,P
      DOUBLE PRECISION COR,EPS
C
C CALCUL DE LA MATRICE B A DIAGONALISER
C
      JSUP = (JMAX*(JMAX+1))/2
      DO 110 J =1,JSUP
  110 B(J) = 0.D0
      DO 111 K =1,JMAX
      DO 112 L=1,K
      JJ = (K*(K-1))/2+L
      DO 113 I=1,IMAX
      B(JJ) = B(JJ) + (P(I,K)*P(I,L))/(P1I(I)*DSQRT(P1J(K)*P1J(L)
  113 CONTINUE
  112 CONTINUE
  111 CONTINUE
C
C TEST
C
      TEST = -1.D0
      DO 200 J=1,JMAX
      J1 = (J*(J-1))/2+J
  200 TEST = TEST + B(J1)
      TEST = TEST *P1
      NR = NB + 1
C
C  DIAGONALISATION DE B
C
      CALL DIAG(JMAX,NR,B,VAL,W,NBIT)
C
C  CALCUL DES FACTEURS
C
       NK = NBIT + 1
      DO 300 M = 2,NR
      IF (VAL(M).GT.0.D0) GO TO 301
      WRITE (6,302)M
      GO TO 9999
  301 M1 =M-1
      VAL2 =DSQRT(VAL(M) )
      M1DEP = (M1-1)*JMAX
      DO 303 J =1,JMAX
         MDEP = (J-1)*NK + M
      M1J = M1DEP + J
      F(M1J) = W(MDEP)/DSQRT(P1J(J)/P1)
      F(M1J) = F(M1J)*VAL2
  303 CONTINUE
      M2DEP = (M1-1)*IMAX
      DO 305 I=1,IMAX
      M1I = M2DEP +I
      IDEP = (I-1)*JMAX
      G(M1I) = 0.D0
      DO 306 J =1,JMAX
```

```
0047          306 G(M1I) = G(M1I) + P(I,J)*F(J+M1DEP)
0048              G(M1I) = G(M1I)/(P1I(I)*VAL2)
0049          305 CONTINUE
0050              IF (M1-1) 307,307,308
0051          307 ATEST(M1) =TEST - P1*VAL(M)
0052              POURC(M1) = ((TEST -ATEST(M1))*100.D00)/TEST
0053              GO TO 309
0054          308 ATEST(M1) = ATEST(M1-1)-P1*VAL(M)
0055              POUPC (M1) = ((ATEST(M1-1)-ATEST(M1))*100.D00)/TEST
0056          309 CONTINUE
0057          302 FORMAT (2H0,I4,23H IEME VALEUR PROPRE NEGATIVE )
0058          700 FORMAT (1X,10E12.5)
                C
                C  CALCUL DE COR%N<
                C
0059                DO 501 N=1,NMAX
0060                COR(N) = 1.D00
0061                I = IT(N)
0062                J = JT(N)
0063                DO 500 M =2,NR
0064                VAL2=DSQRT(VAL(M))
0065                VAL2 =1.D00/VAL2
0066                IM = (M-2)*IMAX
0067                JM = (M-2)*JMAX
0068                IDEP = I + IM
0069                JDEP = J + JM
0070                COR(N) = COR(N) + VAL2*F(JDEP)*G(IDEP)
0071          500 CONTINUE
0072          501 CONTINUE
0073         9999 CONTINUE
0074              RETURN
0075              END
```

```
      SUBROUTINE DIAG(IMAX,NB,Q,VLP,W,NBIT)
      DIMENSION S(20 ),W(120),VLP(6),Q(21  )
      DOUBLE PRECISION Q,VLP,W,S,VNORM,T,VPI,COSI,ALFA,PREC,U
      PREC = 1.D-7
      ITMAX = 50
      NR = NBIT+1
      DO 407 I =1,NB
  407 VLP(I) = 0.D0
      DO 420 NU = 1,NB
      L =0.D00
      MU = 0
      VNORM = 0.D00
      DO 132 I =1,IMAX
  132 S(I) = 0.D0
   20 DO 14 I=1,IMAX
      DO 15 J=1,I
      M = (I*(I-1))/2 + J
      T = Q(M)
      JDEP=(J-1)*NR
   15 S(I) = S(I)+T*W(JDEP+NU)
      I1 =I+1
      IF (I1.GT.IMAX) GO TO 125
      DO 115 J=I1,IMAX
      L=(J*(J-1))/2+I
      JDEP= (J-1)*NP
  115 S(I)=S(I)+Q(L)*W(JDEP+NU)
  125 VNORM =VNORM + S(I)*S(I)
   14 CONTINUE
      VPI =DSQRT(VNORM)
      IF (VPI.GT.1.D-6) GO TO 140
      WRITE(6,1005)NB
      GO TO 468
C
C  CALCUL DE L'ANGLE ET DU TEST
C
  140 COSI = 0.D0
      DO 16 I=1,IMAX
      IDEP =(I-1)*NP
      S(I) = S(I)/VPI
      COSI = COSI + W(IDEP+NU)*S(I)
   16 W(IDEP+NU) = S(I)
      VPI =DSIGN(VPI,COSI)
      IF(DABS(COSI).GT.1.D00) COSI =1.D00
      MU = MU+1
      ALFA =DSQRT(1.D00-COSI*COSI)
      IF (MU.NE.ITMAX) GO TO 21
      NUP = NU-1
      WRITE (6,139)NUP
      GO TO 19
   21 IF (ALFA-PREC) 19,19,18
   18 U = VPI
      VNORM = 0.D00
      DO 22 I=1,IMAX
   22 S(I) = 0.D0
      GO TO 20
   19 VLP(NU) = VPI
C
C CORRECTION DE LA MATRICE
```

```
      C
054         IF (NU-NB) 405,420,420
055       405 DO 116 I=1,IMAX
056           IDEP= (I-1)*NR
057           DO 117 J=1,I
058           JDEP = (J-1)*NR
059           M=(I*(I-1))/2+J
060       117 Q(M) = Q(M)-VPI*W(IDEP+NU)*W(JDEP+NU)
061       116 CONTINUE
062       420 CONTINUE
063         2 FORMAT (16H VALEUR PROPRE #,E12.5,/(10E12.3))
064         3 FORMAT (2X,I3,11H ITERATIONS)
065         4 FORMAT (1H1)
066       139 FORMAT (46H DEFLATION ARRETEE A ITMAX ITERATIONS POUR LE ,I2
           1 12H EME VECTEUR)
067       145 FORMAT (44H  LA MATRICE SEMBLE ETRE DE RANG INFERIEUR A,I3)
068       408 RETURN
069           END
```

Ce programme est celui précédemment nommé par tour, avec remise à zéro.
Les modifications suivantes interviennent dans le sous-programme ITER;
Pour l'avoir par point, retirer les lignes 18,19,20,21,25 et mettre la ligne 17 avant la ligne 26.

Pour ne pas remettre à zéro tour par tour ou point par point, il suffit d'ôter la ligne 16.

De plus, si on veut à chaque rappel du sous-programme repartir de zéro pour chaque p(i,j), après la ligne 7, il faut ajouter :

```
      DØUBLE PRECISION X
      DØ300 N = 1,NMAX
      I = IT(N)
      J = JT(N)
      X = P(I,J)
      P1I(I) = P1I(I)-X
      P1J(J) = P1J(J)-X
      P1 = P1-X
      P(I,J) = 0.DOO
300   CØNTINUE
```

Ce programme peut être rendu plus rapide et plus précis en remplaçant le sous-programme de diagonalisation (DIAG) qui utilise l'algorithme de la puissance itérée et en le remplaçant, par exemple, par celui de Golub et Reinsch.

3 - Tableauxde données utilisées

MU

	*	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	*	15.60	23.90	4.51	13.00	5.11	8.96	20.82	2.04	19.85	27.08	34.05	7.40	25.28	11.47
2	*	15.30	23.99	4.02	13.25	5.15	8.88	20.88	2.03	19.80	27.12	34.09	7.40	25.34	11.77
3	*	15.00	24.12	3.23	13.25	5.16	8.90	20.90	1.96	19.92	27.14	34.14	7.45	25.37	11.69
4	*	14.80	24.35	3.17	13.19	5.17	8.94	21.00	1.96	19.80	27.18	34.18	7.45	25.42	11.96
5	*	14.80	24.55	3.21	13.00	5.18	8.98	21.02	1.90	19.85	27.21	34.23	7.55	25.47	11.90
6	*	14.80	24.87	3.01	13.04	5.17	8.94	21.07	1.89	19.90	27.26	34.24	7.70	25.30	11.73
7	*	14.80	24.67	3.01	13.00	5.18	8.08	21.05	1.33	19.80	27.32	34.26	7.70	25.25	11.75
8	*	14.80	25.74	3.34	13.06	5.19	8.68	21.10	1.79	19.85	27.36	34.30	7.75	25.12	11.79
9	*	14.80	26.65	3.82	13.16	5.18	8.74	21.20	1.89	19.90	27.40	34.35	7.80	25.02	11.76
10	*	15.95	27.23	3.91	13.13	5.18	8.65	21.22	2.10	19.85	27.46	34.41	7.80	25.20	11.77
11	*	14.90	27.67	3.97	13.17	5.18	8.71	21.25	2.14	19.85	27.48	34.42	7.90	25.30	11.36
12	*	14.96	27.94	3.26	13.20	5.17	8.67	21.27	2.10	19.85	27.54	34.46	7.95	25.45	11.12
13	*	14.95	27.25	3.43	13.24	5.17	9.61	21.31	2.16	19.90	27.60	34.49	8.00	25.60	10.50
14	*	15.00	27.58	3.43	13.24	5.17	8.70	21.34	2.09	19.80	27.41	34.51	8.00	25.70	10.39
15	*	15.00	27.84	3.45	13.21	5.15	8.54	21.38	2.15	19.90	27.62	34.55	8.15	25.90	9.93
16	*	15.10	28.10	3.47	13.30	5.16	8.57	21.40	2.26	19.85	27.71	34.57	8.16	25.95	9.37
17	*	15.20	28.31	3.49	13.25	5.18	8.64	21.42	2.30	19.80	27.81	34.59	8.17	25.94	8.94
18	*	15.35	28.52	3.49	13.35	5.17	8.59	21.45	2.38	19.90	27.90	32.60	8.18	26.02	8.94
19	*	15.50	28.90	3.54	13.32	5.16	8.63	21.47	2.45	19.80	27.89	34.63	8.21	25.54	8.47
20	*	15.60	29.03	3.64	13.38	5.12	8.56	21.51	2.53	19.85	28.05	34.68	8.28	25.50	8.12

CI

	*	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	*	15.00	65.00	13.00	7.00	20.00	50.00	28.00	2.00	22.00	55.00	22.00	1.00	29.00	61.00	10.00
2	*	12.00	48.00	25.00	15.00	27.00	26.00	27.00	20.00	7.00	40.00	32.00	21.00	18.00	47.00	35.00
3	*	7.00	30.00	45.00	18.00	7.00	33.00	46.00	14.00	3.00	23.00	55.00	19.00	0.0	37.00	63.00
4	*	14.00	57.00	25.00	4.00	19.00	35.00	34.00	12.00	20.00	48.00	29.00	3.00	16.00	52.00	32.00
5	*	16.00	64.00	10.00	10.00	7.00	41.00	43.00	9.00	7.00	53.00	37.00	3.00	14.00	63.00	18.00
6	*	6.00	39.00	37.00	18.00	7.00	34.00	35.00	24.00	4.00	35.00	51.00	10.00	5.00	58.00	37.00
7	*	33.00	49.00	15.00	3.00	13.00	52.00	27.00	8.00	15.00	62.00	22.00	1.00	37.00	48.00	15.00
8	*	12.00	46.00	28.00	14.00	11.00	37.00	32.00	20.00	3.00	27.00	46.00	24.00	3.00	25.00	72.00
9	*	5.00	37.00	45.00	13.00	10.00	29.00	42.00	19.00	1.00	31.00	55.00	13.00	14.00	52.00	34.00
10	*	4.00	36.00	34.00	26.00	10.00	32.00	28.00	30.00	2.00	17.00	59.00	22.00	0.0	18.00	82.00
11	*	40.00	51.00	7.00	2.00	18.00	36.00	35.00	11.00	28.00	52.00	19.00	1.00	45.00	40.00	15.00
12	*	11.00	60.00	25.00	4.00	9.00	39.00	40.00	12.00	8.00	33.00	47.00	12.00	7.00	68.00	25.00
13	*	12.00	62.00	16.00	10.00	5.00	41.00	42.00	12.00	3.00	46.00	47.00	3.00	9.00	69.00	22.00
14	*	44.00	52.00	5.00	1.00	22.00	52.00	23.00	3.00	46.00	45.00	7.00	2.00	56.00	32.00	12.00
15	*	13.00	49.00	15.00	13.00	14.00	34.00	28.00	24.00	4.00	31.00	54.00	11.00	16.00	59.00	25.00
16	*	0.20	2.70	2.70	4.40	1.20	3.70	3.20	1.90	0.30	2.20	4.60	2.90	1.00	5.00	4.00
17	*	16.00	63.00	15.00	6.00	14.00	43.00	30.00	13.00	32.00	36.00	28.00	4.00	29.00	40.00	36.00
18	*	7.00	34.00	34.00	25.00	5.00	32.00	39.00	24.00	5.00	21.00	62.00	12.00	16.00	46.00	38.00
19	*	6.00	57.00	30.00	7.00	3.00	32.00	39.00	26.00	5.00	33.00	53.00	9.00	6.00	51.00	43.00
20	*	3.00	19.00	45.00	33.00	11.00	40.00	30.00	19.00	3.00	20.00	56.00	21.00	3.00	26.00	71.00
21	*	1.00	10.00	45.00	44.00	5.00	23.00	37.00	35.00	1.00	12.00	65.00	22.00	2.00	17.00	81.00
22	*	6.00	33.00	33.00	28.00	16.00	29.00	33.00	22.00	0.0	21.00	67.00	12.00	6.00	51.00	43.00
23	*	12.00	26.00	35.00	27.00	14.00	29.00	43.00	14.00	10.00	21.00	48.00	21.00	11.00	40.00	49.00
24	*	3.00	24.00	58.00	15.00	4.00	26.00	41.00	29.00	3.00	32.00	56.00	9.00	2.00	43.00	55.00
25	*	2.00	24.00	49.00	25.00	5.00	33.00	32.00	30.00	1.00	18.00	61.00	20.00	4.00	31.00	65.00
26	*	1.00	11.00	43.00	45.00	4.00	20.00	41.00	35.00	0.0	10.00	59.00	21.00	1.00	25.00	74.00
27	*	17.00	55.00	21.00	7.00	14.00	37.00	37.00	12.00	21.00	42.00	33.00	4.00	16.00	58.00	26.00
28	*	2.00	35.00	42.00	21.00	6.00	28.00	43.00	24.00	2.00	23.00	52.00	23.00	2.00	44.00	54.00
29	*	12.00	62.00	27.00	8.00	10.00	46.00	38.00	6.00	13.00	69.00	17.00	1.00	24.00	60.00	11.00
30	*	6.00	29.00	38.00	27.00	21.00	29.00	33.00	17.00	11.00	42.00	30.00	17.00	15.00	31.00	54.00

CA

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	330.0	95.0	360.0	51.0	135.0	20.0	43.0	20.0	97.0	117.0	3.0	1.0	77.0	660.0
2	210.0	500.0	340.0	36.0	40.0	13.0	26.0	21.0	66.0	58.0	3.0	1.0	45.0	460.0
3	260.0	285.0	350.0	44.0	60.0	14.0	24.0	13.0	70.0	70.0	4.0	1.0	54.0	500.0
4	305.0	375.0	375.0	56.0	20.0	3.0	5.0	5.0	5.0	10.0	2.0	1.0	20.0	135.0
5	235.0	290.0	365.0	44.0	20.0	3.0	5.0	5.0	6.0	12.0	2.0	1.0	20.0	120.0
6	135.0	730.0	390.0	32.0	20.0	6.0	15.0	15.0	28.0	21.0	2.0	1.0	40.0	230.0
7	135.0	630.0	325.0	28.0	20.0	8.0	17.0	17.0	42.0	31.0	2.0	1.0	56.0	280.0
8	320.0	230.0	380.0	38.0	105.0	17.0	33.0	14.0	80.0	90.0	3.0	2.0	70.0	440.0
9	305.0	285.0	370.0	40.0	110.0	12.0	26.0	15.0	72.0	70.0	4.0	2.0	60.0	440.0
10	340.0	160.0	460.0	33.0	95.0	8.0	16.0	8.0	56.0	58.0	4.0	1.0	73.0	360.0
11	340.0	130.0	660.0	31.0	160.0	12.0	18.0	12.0	58.0	52.0	4.0	1.0	58.0	250.0
12	380.0	180.0	550.0	42.0	230.0	13.0	27.0	16.0	73.0	61.0	5.0	1.0	90.0	440.0
13	280.0	180.0	580.0	27.0	75.0	10.0	22.0	9.0	57.0	65.0	3.0	1.0	70.0	500.0
14	150.0	335.0	190.0	18.0	20.0	24.0	9.0	5.0	15.0	11.0	1.0	1.0	45.0	65.0
15	295.0	350.0	370.0	35.0	85.0	11.0	24.0	13.0	70.0	63.0	4.0	1.0	64.0	500.0
16	315.0	180.0	445.0	57.0	180.0	24.0	40.0	17.0	104.0	121.0	3.0	1.0	93.0	540.0
17	310.0	150.0	350.0	50.0	100.0	51.0	34.0	10.0	90.0	104.0	3.0	1.0	71.0	460.0
18	315.0	240.0	350.0	47.0	110.0	30.0	34.0	14.0	91.0	105.0	4.0	1.0	74.0	440.0
19	290.0	250.0	290.0	46.0	85.0	76.0	28.0	12.0	77.0	81.0	4.0	1.0	64.0	440.0
20	215.0	330.0	355.0	28.0	20.0	5.0	14.0	13.0	36.0	28.0	3.0	1.0	90.0	200.0
21	205.0	350.0	325.0	25.0	20.0	6.0	13.0	8.0	54.0	34.0	3.0	1.0	27.0	500.0
22	300.0	380.0	335.0	37.0	50.0	9.0	20.0	16.0	55.0	50.0	5.0	3.0	56.0	460.0
23	345.0	270.0	390.0	45.0	160.0	15.0	31.0	20.0	77.0	77.0	3.0	1.0	73.0	460.0
24	210.0	550.0	465.0	47.0	30.0	7.0	14.0	13.0	33.0	36.0	3.0	1.0	28.0	330.0
25	280.0	400.0	370.0	35.0	65.0	11.0	20.0	13.0	62.0	51.0	4.0	2.0	40.0	540.0
26	305.0	250.0	460.0	35.0	105.0	24.0	33.0	19.0	85.0	86.0	5.0	3.0	72.0	640.0
27	320.0	260.0	370.0	38.0	170.0	27.0	20.0	13.0	61.0	53.0	3.0	1.0	57.0	460.0
28	312.0	340.0	535.0	28.0	105.0	8.0	17.0	11.0	59.0	50.0	3.0	1.0	41.0	320.0
29	335.0	180.0	705.0	26.0	145.0	6.0	16.0	10.0	48.0	47.0	3.0	1.0	53.0	320.0
30	360.0	200.0	375.0	29.0	145.0	7.0	16.0	7.0	55.0	49.0	2.0	1.0	51.0	400.0
31	370.0	220.0	420.0	38.0	200.0	14.0	26.0	14.0	77.0	84.0	4.0	1.0	72.0	560.0
32	72.0	999.0	460.0	21.0	20.0	3.0	5.0	5.0	5.0	15.0	1.0	1.0	20.0	220.0
33	450.0	520.0	730.0	27.0	150.0	12.0	29.0	20.0	60.0	67.0	2.0	1.0	39.0	540.0
34	195.0	500.0	260.0	35.0	20.0	12.0	6.0	5.0	10.0	10.0	1.0	1.0	20.0	150.0
35	130.0	400.0	305.0	6.0	20.0	3.0	9.0	5.0	15.0	11.0	1.0	1.0	20.0	240.0
36	315.0	380.0	480.0	57.0	20.0	3.0	5.0	5.0	5.0	17.0	2.0	1.0	20.0	220.0
37	150.0	570.0	555.0	22.0	45.0	16.0	15.0	11.0	25.0	40.0	2.0	1.0	24.0	260.0
38	150.0	730.0	325.0	26.0	55.0	11.0	21.0	12.0	48.0	48.0	2.0	1.0	50.0	360.0
39	72.0	999.0	510.0	13.0	20.0	3.0	5.0	5.0	5.0	10.0	1.0	1.0	35.0	240.0
40	320.0	220.0	390.0	29.0	95.0	6.0	13.0	5.0	55.0	54.0	3.0	1.0	54.0	440.0
41	140.0	800.0	350.0	26.0	20.0	9.0	27.0	14.0	41.0	33.0	2.0	1.0	50.0	460.0
42	310.0	270.0	460.0	35.0	80.0	10.0	19.0	14.0	65.0	50.0	5.0	1.0	71.0	260.0
43	270.0	400.0	480.0	41.0	125.0	12.0	25.0	18.0	80.0	67.0	2.0	1.0	76.0	440.0
44	345.0	220.0	455.0	50.0	145.0	17.0	34.0	15.0	82.0	105.0	3.0	1.0	82.0	640.0
45	150.0	760.0	420.0	17.0	30.0	9.0	23.0	17.0	40.0	36.0	1.0	1.0	31.0	260.0
46	175.0	620.0	310.0	17.0	70.0	11.0	23.0	11.0	40.0	40.0	2.0	1.0	33.0	300.0
47	200.0	370.0	150.0	25.0	20.0	4.0	10.0	10.0	12.0	10.0	1.0	1.0	20.0	120.0
48	110.0	870.0	395.0	20.0	20.0	7.0	15.0	15.0	21.0	16.0	1.0	1.0	20.0	180.0
49	340.0	200.0	405.0	46.0	150.0	20.0	36.0	19.0	100.0	115.0	4.0	2.0	92.0	660.0
50	320.0	160.0	410.0	63.0	155.0	29.0	42.0	19.0	108.0	133.0	3.0	1.0	87.0	840.0

51	*	40.0	999.0	481.0	6.0	20.0	46.0	14.0	5.0	5.0	10.0	1.0	1.0	20.0	170.0
52	*	310.0	999.0	250.0	6.0	20.0	27.0	5.0	5.0	5.0	52.0	2.0	1.0	20.0	400.0
53	*	100.0	940.0	430.0	12.0	20.0	7.0	15.0	13.0	24.0	20.0	1.0	1.0	20.0	150.0
54	*	300.0	170.0	350.0	28.0	40.0	3.0	7.0	5.0	28.0	23.0	3.0	1.0	68.0	550.0
55	*	290.0	110.0	315.0	47.0	20.0	3.0	5.0	5.0	6.0	18.0	1.0	1.0	20.0	50.0
56	*	270.0	250.0	370.0	34.0	50.0	10.0	24.0	12.0	71.0	67.0	4.0	1.0	53.0	480.0
57	*	300.0	180.0	395.0	55.0	85.0	22.0	36.0	13.0	97.0	108.0	2.0	1.0	90.0	440.0
58	*	345.0	210.0	225.0	30.0	55.0	5.0	13.0	8.0	33.0	34.0	2.0	1.0	81.0	440.0
59	*	250.0	320.0	260.0	23.0	20.0	6.0	13.0	6.0	43.0	39.0	3.0	1.0	46.0	460.0
60	*	280.0	160.0	395.0	44.0	20.0	3.0	5.0	5.0	6.0	16.0	1.0	1.0	20.0	75.0
61	*	285.0	120.0	255.0	42.0	20.0	3.0	5.0	5.0	5.0	13.0	1.0	1.0	20.0	85.0
62	*	315.0	95.0	230.0	42.0	20.0	3.0	5.0	5.0	5.0	13.0	2.0	1.0	50.0	85.0
63	*	190.0	420.0	200.0	24.0	20.0	4.0	13.0	12.0	27.0	26.0	2.0	1.0	33.0	340.0
64	*	310.0	310.0	150.0	58.0	20.0	9.0	5.0	5.0	5.0	10.0	2.0	1.0	20.0	130.0
65	*	310.0	265.0	265.0	48.0	20.0	3.0	5.0	5.0	5.0	17.0	2.0	1.0	20.0	135.0
66	*	205.0	510.0	440.0	51.0	40.0	9.0	11.0	7.0	22.0	28.0	2.0	1.0	44.0	270.0
67	*	200.0	999.0	825.0	26.0	20.0	3.0	5.0	5.0	5.0	37.0	1.0	1.0	28.0	200.0
68	*	220.0	670.0	990.0	43.0	20.0	6.0	8.0	9.0	12.0	32.0	2.0	1.0	20.0	170.0
69	*	250.0	200.0	355.0	48.0	20.0	3.0	8.0	5.0	6.0	19.0	2.0	1.0	20.0	120.0
70	*	230.0	450.0	295.0	31.0	90.0	10.0	21.0	9.0	57.0	60.0	3.0	1.0	70.0	460.0
71	*	330.0	350.0	450.0	41.0	200.0	16.0	30.0	14.0	80.0	88.0	3.0	1.0	90.0	540.0
72	*	270.0	590.0	360.0	25.0	50.0	15.0	22.0	14.0	57.0	47.0	2.0	1.0	55.0	440.0
73	*	430.0	250.0	345.0	43.0	180.0	13.0	29.0	18.0	67.0	57.0	3.0	1.0	60.0	400.0
74	*	390.0	190.0	350.0	31.0	140.0	9.0	17.0	6.0	58.0	63.0	3.0	1.0	43.0	460.0
75	*	325.0	300.0	355.0	33.0	140.0	11.0	27.0	13.0	60.0	57.0	3.0	2.0	43.0	460.0
76	*	445.0	180.0	390.0	34.0	215.0	10.0	22.0	9.0	69.0	78.0	3.0	1.0	50.0	460.0
77	*	325.0	180.0	360.0	27.0	100.0	10.0	16.0	6.0	59.0	67.0	3.0	1.0	52.0	460.0
78	*	370.0	200.0	375.0	36.0	185.0	9.0	16.0	8.0	53.0	59.0	3.0	1.0	50.0	440.0
79	*	365.0	320.0	540.0	28.0	200.0	12.0	24.0	14.0	56.0	49.0	2.0	1.0	48.0	340.0
80	*	175.0	999.0	510.0	25.0	30.0	6.0	5.0	5.0	5.0	26.0	1.0	1.0	34.0	220.0
81	*	220.0	600.0	400.0	25.0	65.0	6.0	20.0	12.0	45.0	41.0	3.0	1.0	43.0	360.0
82	*	325.0	190.0	620.0	32.0	105.0	11.0	14.0	10.0	56.0	46.0	5.0	1.0	67.0	340.0
83	*	240.0	360.0	220.0	41.0	25.0	3.0	8.0	7.0	12.0	10.0	1.0	1.0	20.0	150.0
84	*	320.0	280.0	415.0	47.0	135.0	20.0	32.0	16.0	86.0	88.0	4.0	1.0	70.0	340.0
85	*	185.0	590.0	520.0	25.0	20.0	3.0	14.0	14.0	34.0	19.0	2.0	1.0	40.0	240.0
86	*	320.0	200.0	540.0	26.0	125.0	8.0	14.0	8.0	54.0	43.0	4.0	1.0	55.0	360.0
87	*	265.0	330.0	200.0	42.0	60.0	11.0	16.0	10.0	34.0	24.0	2.0	1.0	41.0	440.0
88	*	160.0	999.0	560.0	34.0	20.0	3.0	51.0	5.0	29.0	10.0	2.0	1.0	20.0	170.0
89	*	165.0	200.0	230.0	22.0	20.0	8.0	5.0	5.0	5.0	10.0	2.0	1.0	20.0	60.0
90	*	390.0	180.0	440.0	33.0	160.0	8.0	20.0	11.0	56.0	49.0	3.0	1.0	56.0	460.0
91	*	400.0	200.0	450.0	32.0	165.0	10.0	21.0	11.0	65.0	56.0	1.0	1.0	54.0	360.0
92	*	155.0	500.0	275.0	21.0	70.0	12.0	25.0	16.0	47.0	31.0	2.0	1.0	33.0	360.0
93	*	380.0	175.0	415.0	43.0	200.0	14.0	25.0	13.0	63.0	60.0	2.0	1.0	50.0	340.0
94	*	290.0	70.0	400.0	61.0	290.0	20.0	31.0	5.0	80.0	96.0	3.0	1.0	55.0	850.0
95	*	420.0	170.0	560.0	47.0	340.0	22.0	36.0	15.0	96.0	102.0	3.0	1.0	87.0	560.0
96	*	380.0	200.0	360.0	50.0	240.0	19.0	31.0	17.0	88.0	98.0	3.0	1.0	74.0	380.0
97	*	370.0	150.0	410.0	65.0	400.0	24.0	33.0	15.0	93.0	113.0	3.0	1.0	90.0	550.0
98	*	365.0	170.0	405.0	47.0	220.0	14.0	25.0	8.0	74.0	78.0	2.0	1.0	65.0	440.0
99	*	410.0	210.0	450.0	28.0	145.0	9.0	18.0	8.0	59.0	59.0	2.0	1.0	42.0	440.0
100	*	370.0	190.0	400.0	29.0	140.0	10.0	18.0	11.0	60.0	48.0	2.0	1.0	44.0	320.0
101	*	280.0	370.0	450.0	27.0	110.0	13.0	17.0	14.0	55.0	64.0	3.0	1.0	47.0	340.0
102	*	290.0	210.0	570.0	27.0	100.0	8.0	29.0	14.0	70.0	71.0	3.0	1.0	60.0	400.0
103	*	355.0	150.0	425.0	39.0	170.0	18.0	19.0	9.0	65.0	71.0	3.0	1.0	56.0	460.0
104	*	360.0	170.0	460.0	38.0	220.0	12.0	19.0	10.0	66.0	75.0	3.0	1.0	54.0	400.0
105	*	345.0	130.0	340.0	64.0	300.0	39.0	31.0	9.0	91.0	94.0	3.0	1.0	90.0	700.0
106	*	165.0	500.0	260.0	23.0	40.0	17.0	26.0	11.0	63.0	65.0	4.0	2.0	70.0	320.0
107	*	275.0	350.0	395.0	34.0	110.0	15.0	77.0	15.0	83.0	100.0	3.0	1.0	85.0	740.0
108	*	245.0	340.0	310.0	31.0	57.0	48.0	29.0	11.0	69.0	69.0	3.0	1.0	82.0	640.0
109	*	205.0	460.0	340.0	37.0	50.0	17.0	41.0	20.0	94.0	116.0	3.0	1.0	90.0	700.0
110	*	270.0	300.0	370.0	46.0	40.0	15.0	35.0	15.0	85.0	105.0	3.0	1.0	78.0	380.0
111	*	240.0	400.0	330.0	36.0	60.0	11.0	22.0	12.0	68.0	67.0	3.0	1.0	80.0	500.0
112	*	295.0	300.0	390.0	33.0	55.0	12.0	35.0	14.0	90.0	93.0	4.0	1.0	82.0	700.0
113	*	150.0	600.0	270.0	20.0	20.0	10.0	24.0	12.0	60.0	51.0	3.0	1.0	53.0	540.0
114	*	235.0	999.0	990.0	38.0	20.0	3.0	5.0	5.0	5.0	30.0	2.0	1.0	28.0	200.0
115	*	315.0	140.0	90.0	32.0	20.0	3.0	5.0	5.0	12.0	10.0	2.0	1.0	22.0	340.0
116	*	230.0	190.0	160.0	30.0	20.0	3.0	5.0	5.0	22.0	15.0	2.0	1.0	88.0	260.0
117	*	175.0	90.0	70.0	20.0	20.0	3.0	5.0	5.0	5.0	10.0	1.0	1.0	20.0	60.0
118	*	280.0	280.0	60.0	29.0	20.0	3.0	5.0	5.0	16.0	10.0	1.0	1.0	20.0	125.0

PR

	*	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	*	349.00	119.00	52.00	164.00	119.00	53.00	50.00	42.00	54.00	34.00	89.00	88.00	28.00	87.00	66.00	66.00	60.00	137.00	156.00	30.0
2	*	265.00	79.00	42.00	129.00	105.00	44.00	45.00	40.00	40.00	28.00	75.00	77.00	19.00	63.00	55.00	56.00	47.00	97.00	136.00	23.0
3	*	278.00	91.00	46.00	141.00	113.00	49.00	47.00	43.00	45.00	28.00	82.00	82.00	20.00	61.00	58.00	59.00	49.00	102.00	119.00	26.0
4	*	289.00	95.00	46.00	142.00	112.00	47.00	46.00	42.00	47.00	30.00	81.00	85.00	20.00	66.00	55.00	61.00	50.00	109.00	138.00	24.0
5	*	276.00	89.00	46.00	140.00	107.00	46.00	46.00	38.00	43.00	30.00	82.00	81.00	20.00	70.00	53.00	59.00	50.00	117.00	122.00	20.0
6	*	296.00	100.00	49.00	145.00	112.00	49.00	49.00	42.00	49.00	34.00	83.00	81.00	23.00	74.00	55.00	65.00	52.00	127.00	145.00	27.0
7	*	321.00	110.00	52.00	156.00	114.00	48.00	47.00	46.00	50.00	36.00	89.00	82.00	24.00	75.00	63.00	68.00	57.00	128.00	147.00	25.0
8	*	319.00	116.00	48.00	154.00	119.00	49.00	49.00	40.00	48.00	29.00	88.00	84.00	23.00	78.00	62.00	63.00	60.00	121.00	151.00	26.0
9	*	292.00	96.00	47.00	150.00	114.00	47.00	46.00	40.00	43.00	29.00	78.00	81.00	22.00	72.00	55.00	59.00	52.00	123.00	133.00	23.0
10	*	304.00	101.00	47.00	148.00	113.00	47.00	48.00	42.00	49.00	31.00	84.00	82.00	24.00	71.00	62.00	65.00	55.00	120.00	151.00	26.0
11	*	325.00	111.00	50.00	161.00	120.00	50.00	49.00	46.00	53.00	35.00	87.00	83.00	25.00	78.00	64.00	68.00	57.00	126.00	146.00	27.0
12	*	284.00	93.00	46.00	143.00	114.00	49.00	47.00	42.00	46.00	30.00	80.00	76.00	21.00	66.00	60.00	62.00	52.00	108.00	133.00	26.0
13	*	259.00	81.00	44.00	130.00	104.00	45.00	45.00	40.00	40.00	27.00	78.00	78.00	18.00	61.00	51.00	57.00	46.00	110.00	130.00	24.0
14	*	275.00	89.00	47.00	141.00	111.00	45.00	44.00	43.00	48.00	34.00	84.00	84.00	22.00	65.00	52.00	61.00	50.00	117.00	126.00	24.0
15	*	299.00	95.00	47.00	143.00	115.00	51.00	49.00	44.00	44.00	30.00	81.00	83.00	22.00	69.00	57.00	63.00	54.00	117.00	148.00	26.0
16	*	323.00	114.00	50.00	165.00	119.00	49.00	47.00	40.00	51.00	34.00	85.00	87.00	24.00	79.00	61.00	65.00	58.00	125.00	133.00	25.0
17	*	312.00	104.00	47.00	154.00	117.00	50.00	50.00	45.00	49.00	34.00	83.00	76.00	24.00	73.00	61.00	66.00	54.00	125.00	145.00	27.0
18	*	307.00	101.00	49.00	154.00	114.00	44.00	44.00	40.00	51.00	33.00	86.00	85.00	24.00	69.00	59.00	63.00	54.00	124.00	152.00	25.0
19	*	281.00	88.00	50.00	141.00	111.00	46.00	45.00	39.00	46.00	29.00	81.00	79.00	20.00	70.00	50.00	58.00	48.00	115.00	136.00	23.0
20	*	300.00	95.00	45.00	146.00	111.00	47.00	45.00	37.00	48.00	27.00	81.00	82.00	23.00	70.00	57.00	59.00	56.00	121.00	153.00	25.0
21	*	290.00	95.00	48.00	151.00	111.00	45.00	45.00	36.00	47.00	34.00	82.00	86.00	22.00	73.00	52.00	60.00	55.00	118.00	138.00	25.0
22	*	314.00	102.00	51.00	153.00	122.00	53.00	50.00	45.00	50.00	32.00	84.00	82.00	24.00	69.00	64.00	63.00	55.00	130.00	155.00	27.0
23	*	267.00	97.00	51.00	152.00	119.00	45.00	45.00	39.00	52.00	31.00	83.00	84.00	23.00	69.00	58.00	59.00	53.00	128.00	154.00	26.0
24	*	306.00	103.00	47.00	154.00	115.00	49.00	49.00	44.00	49.00	33.00	85.00	84.00	24.00	73.00	61.00	67.00	54.00	122.00	138.00	27.0
25	*	259.00	82.00	43.00	128.00	104.00	48.00	47.00	40.00	47.00	26.00	77.00	80.00	17.00	64.00	49.00	57.00	46.00	110.00	120.00	25.0
26	*	240.00	76.00	45.00	126.00	102.00	48.00	48.00	46.00	43.00	25.00	75.00	77.00	15.00	61.00	52.00	58.00	40.00	94.00	103.00	24.0
27	*	308.00	103.00	51.00	150.00	117.00	48.00	48.00	43.00	49.00	30.00	87.00	88.00	23.00	72.00	61.00	63.00	53.00	125.00	156.00	27.0
28	*	307.00	100.00	48.00	151.00	119.00	49.00	48.00	46.00	52.00	31.00	83.00	85.00	22.00	76.00	57.00	65.00	54.00	124.00	141.00	27.0
29	*	315.00	103.00	47.00	156.00	118.00	46.00	46.00	43.00	52.00	32.00	86.00	86.00	23.00	76.00	61.00	65.00	53.00	124.00	156.00	26.0
30	*	317.00	107.00	51.00	157.00	120.00	48.00	48.00	45.00	50.00	35.00	88.00	86.00	23.00	81.00	57.00	66.00	60.00	124.00	141.00	26.0
31	*	301.00	100.00	51.00	154.00	117.00	50.00	49.00	45.00	48.00	33.00	84.00	84.00	22.00	78.00	59.00	65.00	52.00	123.00	146.00	27.0
32	*	288.00	99.00	47.00	152.00	112.00	47.00	45.00	42.00	50.00	33.00	84.00	83.00	23.00	75.00	57.00	65.00	53.00	120.00	150.00	26.0
33	*	317.00	106.00	49.00	153.00	118.00	47.00	48.00	42.00	50.00	35.00	86.00	83.00	23.00	78.00	58.00	64.00	54.00	120.00	143.00	25.0
34	*	230.00	68.00	52.00	156.00	107.00	49.00	49.00	42.00	49.00	33.00	86.00	89.00	21.00	73.00	58.00	66.00	54.00	116.00	134.00	26.0
35	*	299.00	100.00	46.00	145.00	113.00	47.00	47.00	41.00	45.00	28.00	86.00	84.00	22.00	76.00	59.00	60.00	52.00	113.00	144.00	25.0
36	*	304.00	98.00	47.00	150.00	115.00	51.00	50.00	42.00	50.00	32.00	83.00	84.00	22.00	74.00	58.00	64.00	54.00	122.00	147.00	26.0
37	*	319.00	110.00	50.00	159.00	120.00	47.00	47.00	47.00	51.00	33.00	87.00	84.00	26.00	78.00	60.00	66.00	56.00	123.00	147.00	26.0
38	*	304.00	103.00	47.00	151.00	115.00	47.00	47.00	41.00	50.00	33.00	84.00	82.00	24.00	74.00	60.00	62.00	54.00	115.00	146.00	26.0
39	*	316.00	110.00	53.00	163.00	119.00	46.00	47.00	46.00	55.00	36.00	89.00	89.00	24.00	72.00	64.00	68.00	59.00	125.00	135.00	25.0
40	*	303.00	101.00	46.00	150.00	113.00	49.00	48.00	42.00	50.00	33.00	81.00	78.00	21.00	77.00	60.00	64.00	57.00	110.00	143.00	25.0
41	*	314.00	106.00	50.00	152.00	118.00	47.00	49.00	41.00	47.00	32.00	87.00	83.00	22.00	69.00	60.00	62.00	55.00	114.00	134.00	27.0
42	*	218.00	66.00	41.00	120.00	95.00	44.00	44.00	44.00	40.00	24.00	71.00	75.00	14.00	59.00	51.00	52.00	38.00	78.00	86.00	21.0
43	*	249.00	79.00	43.00	134.00	106.00	45.00	45.00	38.00	42.00	27.00	74.00	75.00	17.00	65.00	47.00	55.00	44.00	98.00	119.00	23.0
44	*	270.00	92.00	47.00	144.00	116.00	45.00	45.00	45.00	50.00	32.00	83.00	84.00	21.00	70.00	54.00	61.00	52.00	116.00	129.00	26.0
45	*	302.00	100.00	49.00	150.00	119.00	47.00	47.00	45.00	50.00	29.00	84.00	83.00	22.00	69.00	60.00	62.00	55.00	114.00	134.00	27.0
46	*	300.00	100.00	48.00	147.00	118.00	49.00	47.00	44.00	48.00	31.00	87.00	84.00	21.00	74.00	57.00	64.00	52.00	118.00	157.00	27.0
47	*	287.00	92.00	47.00	146.00	110.00	46.00	46.00	41.00	53.00	31.00	81.00	79.00	21.00	70.00	54.00	61.00	54.00	115.00	138.00	25.0
48	*	251.00	79.00	45.00	134.00	109.00	46.00	46.00	45.00	42.00	30.00	75.00	77.00	18.00	63.00	52.00	61.00	45.00	94.00	115.00	24.0
49	*	302.00	97.00	48.00	149.00	110.00	45.00	46.00	37.00	46.00	31.00	82.00	81.00	22.00	65.00	56.00	61.00	58.00	133.00	147.00	27.0
50	*	325.00	110.00	52.00	151.00	122.00	49.00	50.00	46.00	52.00	32.00	84.00	86.00	24.00	84.00	62.00	67.00	57.00	140.00	149.00	28.0
51	*	295.00	96.00	50.00	139.00	111.00	47.00	45.00	38.00	48.00	29.00	81.00	82.00	22.00	72.00	59.00	59.00	55.00	132.00	152.00	27.0
52	*	242.00	87.00	45.00	146.00	111.00	45.00	44.00	42.00	44.00	32.00	80.00	81.00	20.00	69.00	51.00	59.00	52.00	125.00	135.00	26.0
53	*	294.00	95.00	49.00	142.00	112.00	45.00	45.00	41.00	46.00	33.00	82.00	81.00	24.00	72.00	54.00	61.00	52.00	124.00	141.00	27.0
54	*	274.00	88.00	48.00	138.00	103.00	43.00	44.00	39.00	46.00	28.00	81.00	84.00	21.00	63.00	53.00	57.00	49.00	134.00	133.00	26.0
55	*	299.00	99.00	46.00	143.00	111.00	44.00	43.00	40.00	46.00	30.00	81.00	82.00	23.00	73.00	54.00	61.00	55.00	121.00	138.00	25.0
56	*	328.00	106.00	45.00	149.00	116.00	47.00	44.00	43.00	49.00	33.00	85.00	86.00	24.00	76.00	62.00	62.00	55.00	135.00	145.00	26.0

```
* 295.00 96.00 45.00142.00110.00 47.00 47.00 39.00 47.00 31.00 81.00 83.00 22.00 69.00 60.00 61.00 53.00123.00139.00
* 209.00 63.00 41.00111.00 95.00 44.00 45.00 42.00 37.00 23.00 64.00 70.00 13.00 50.00 46.00 53.00 39.00 73.00 94.00
* 259.00 81.00 46.00133.00108.00 44.00 44.00 43.00 41.00 26.00 77.00 82.00 19.00 61.00 48.00 57.00 46.00108.00134.00
* 280.00 89.00 46.00142.00112.00 45.00 44.00 37.00 47.00 29.00 79.00 82.00 21.00 64.00 53.00 [illegible] 53.00130.00147
* 288.00 91.00 49.00139.00109.00 50.00 49.00 44.00 48.00 30.00 78.00 80.00 21.00 70.00 59.00 61.00 52.00112.0014
* 300.00 99.00 50.00149.00116.00 48.00 44.00 45.00 48.00 30.00 85.00 85.00 23.00 69.00 61.00 64.00
* 224.00 70.00 44.00121.00100.00 48.00 49.00 43.00 39.00 25.00 69.00 78.00 14.00 59.00 53.00 [illegible] 39.00 74.00
* 319.00110.00 50.00152.00119.00 49.00 47.00 45.00 52.00 32.00 84.00 84.00 25.00 88.00 57.00 63.00
* 276.00 88.00 47.00134.00111.00 48.00 44.00 44.00 44.00 29.00 79.00 78.00 20.00 68.00 56.00 50.00 50.00112.0012
* 249.00 81.00 43.00123.00104.00 48.00 47.00 42.00 41.00 28.00 72.00 72.00 19.00 64.00 52.00 59.00 43.00 96.00
* 265.00 88.00 46.00130.00108.00 50.00 49.00 46.00 42.00 28.00 76.00 78.00 18.00 67.00 51.00 62.00 45.00103.0011
* 255.00 89.00 47.00122.00108.00 44.00 47.00 50.00 42.00 29.00 77.00 82.00 19.00 68.00 44.00 61.00 47.00102.00113.0
* 325.00108.00 50.00154.00120.00 50.00 48.00 40.00 53.00 34.00 83.00 84.00 26.00 79.00 61.00 64.00 60.00135.00145.00
* 248.00116.00 49.00160.00122.00 46.00 48.00 40.00 55.00 33.00 92.00 86.00 25.00 85.00 68.00 67.00 61.00145.00148.00
* 259.00 82.00 44.00127.00102.00 45.00 44.00 39.00 41.00 27.00 77.00 80.00 19.00 65.00 49.00 54.00 46.00102.00121.00
* 342.00119.00 54.00161.00123.00 48.00 48.00 40.00 57.00 35.00 92.00 91.00 25.00 88.00 64.00 68.00 63.00141.00164.00
* 328.00112.00 53.00161.00122.00 51.00 49.00 47.00 52.00 37.00 89.00 87.00 23.00 75.00 64.00 67.00 60.00131.00151.00
* 291.00 92.00 47.00144.00115.00 50.00 48.00 47.00 52.00 30.00 80.00 82.00 20.00 64.00 60.00 52.00 49.00110.00142.00 28.
* 257.00 83.00 44.00129.00107.00 48.00 50.00 43.00 43.00 25.00 76.00 79.00 16.00 68.00 46.00 58.00 44.00 94.00111 24.
* 238.00 72.00 45.00125.00103.00 47.00 46.00 45.00 45.00 26.00 72.00 75.00 16.00 57.00 49.00 57.00 41.00 98.00107.00 24.
* 299.00 93.00 47.00146.00113.00 47.00 48.00 40.00 51.00 30.00 85.00 85.00 19.00 67.00 57.00 62.00 25.
* 240.00 70.00 46.00127.00105.00 45.00 49.00 45.00 47.00 26.00 76.00 79.00 14.00 58.00 48.00 46.00 42.00100.00110.00 24.
* 243.00 76.00 44.00128.00103.00 45.00 46.00 44.00 47.00 26.00 73.00 71.00 17.00 63.00 47.00 48.00 43.00100.00109.00 24.
* 256.00 80.00 45.00129.00102.00 47.00 47.00 41.00 47.00 30.00 78.00 79.00 18.00 63.00 51.00 60.00 25.
* 274.00 89.00 47.00137.00109.00 50.00 51.00 47.00 50.00 28.00 80.00 82.00 19.00 63.00 56.00 63.00 53.00111.00130.00 28.
* 342.00117.00 54.00165.00122.00 51.00 50.00 46.00 60.00 35.00 89.00 88.00 25.00 77.00 66.00 68.00 64.00147.00152.00 29.
* 329.00110.00 50.00160.00117.00 51.00 50.00 41.00 54.00 32.00 89.00 86.00 24.00 79.00 62.00 66.00 61.00138.00144. 24.
* 318.00107.00 49.00155.00119.00 48.00 44.00 42.00 52.00 31.00 87.00 84.00 25.00 78.00 63.00 62.00 28.
* 313.00101.00 51.00143.00111.00 46.00 46.00 38.00 48.00 31.00 84.00 83.00 22.00 70.00 58.00 61.00 25.
* 287.00 94.00 49.00138.00112.00 49.00 48.00 42.00 43.00 30.00 81.00 81.00 21.00 72.00 55.00 60.00 27.
* 271.00 88.00 47.00135.00108.00 46.00 45.00 40.00 44.00 29.00 79.00 81.00 23.00 69.00 52.00 59.00 25.
* 291.00 92.00 46.00140.00113.00 49.00 47.00 44.00 46.00 32.00 81.00 83.00 22.00 73.00 53.00 63. 27.
* 317.00106.00 50.00148.00116.00 46.00 48.00 42.00 50.00 32.00 85.00 82.00 25.00 79.00 58.00 63.00 28.
* 320.00109.00 51.00151.00119.00 51.00 50.00 48.00 50.00 32.00 87.00 87.00 25.00 79.00 62.00 66.00
* 305.00109.00 50.00152.00116.00 47.00 47.00 37.00 52.00 28.00 84.00 85.00 23.00 73.00 55.00 60.00 55.00126.00149
* 331.00115.00 54.00161.00123.00 50.00 51.00 44.00 51.00 34.00 89.00 87.00 25.00 72.00 66.00 69.00
* 291.00 96.00 44.00138.00118.00 48.00 47.00 41.00 46.00 29.00 82.00 76.00 21.00 70.00 60.00 59.00
* 326.00110.00 50.00156.00119.00 50.00 50.00 46.00 51.00 31.00 86.00 92.00 26.00 79.00 62.00 65.00
* 325.00112.00 53.00156.00123.00 49.00 48.00 43.00 56.00 34.00 85.00 85.00 25.00 80.00 56.00 67.00
* 300.00105.00 51.00152.00117.00 43.00 43.00 36.00 50.00 33.00 85.00 86.00 22.00 72.00 57.00 60.00 52.00121.00130.00
* 314.00103.00 49.00159.00117.00 50.00 48.00 45.00 49.00 32.00 86.00 99.00 23.00 74.00 59.00 64.00 55.00132.00150.00
* 304.00103.00 50.00149.00115.00 47.00 45.00 41.00 47.00 31.00 85.00 83.00 26.00 75.00 59.00 62.00 55.00125.00134.00
* 257.00 82.00 50.00133.00113.00 48.00 47.00 47.00 43.00 23.00 77.00 79.00 18.00 64.00 55.00 62.00 47.00 94.00113.00
* 274.00 86.00 49.00143.00115.00 48.00 43.00 45.00 43.00 31.00 78.00 79.00 20.00 62.00 55.00 62.00 47.00106.00119.00
* 254.00 76.00 45.00124.00105.00 45.00 45.00 39.00 42.00 26.00 76.00 80.00 19.00 61.00 50.00 55.00 45.00104.00129.00
* 298.00 99.00 50.00145.00122.00 49.00 43.00 44.00 50.00 31.00 80.00 92.00 24.00 71.00 57.00 63.00 53.00136.00155.00
* 296.00 99.00 47.00137.00110.00 47.00 47.00 44.00 46.00 30.00 82.00 92.00 22.00 77.00 56.00 62.00 55.00121.00130.00
* 323.00104.00 52.00152.00115.00 50.00 49.00 44.00 51.00 33.00 90.00 83.00 25.00 76.00 61.00 65.00 57.00133.00145.0
* 328.00113.00 54.00153.00120.00 46.00 47.00 40.00 50.00 32.00 93.00 84.00 27.00 91.00 59.00 64.00 [illegible].00138.00142.00
* 302.00102.00 46.00141.00113.00 49.00 47.00 40.00 47.00 32.00 92.00 82.00 24.00 73.00 56.00 62.00 53.00123.00136.00
* 318.00106.00 51.00143.00111.00 50.00 48.00 41.00 51.00 30.00 89.00 86.00 25.00 80.00 53.00 63.00 59.00135.00155.00
* 319.00105.00 53.00157.00113.00 46.00 46.00 42.00 52.00 32.00 91.00 91.00 25.00 71.00 62.00 63.00 59.00124.00144.00
* 296.00 95.00 48.00145.00113.00 47.00 48.00 46.00 44.00 27.00 81.00 82.00 24.00 72.00 57.00 59. [illegible].00121.00144.00
* 295.00 92.00 49.00148.00120.00 50.00 48.00 46.00 44.00 30.00 85.00 89.00 22.00 72.00 60.00 62.00 54.00118.00136.00
* 325.00108.00 52.00159.00122.00 50.00 49.00 43.00 49.00 33.00 88.00 98.00 24.00 80.00 64.00 64.
* 325.00103.00 50.00157.00122.00 49.00 47.00 45.00 49.00 32.00 87.00 83.00 26.00 75.00 59.00 63.00 57.00137.00156.00
* 307.00100.00 49.00152.00110.00 44.00 49.00 44.00 49.00 33.00 84.00 96.00 23.00 72.00 67.00 64.00
* 306.00 99.00 49.00151.00110.00 44.00 49.00 43.00 46.00 34.00 84.00 86.00 23.00 71.00 57.00 64.00 52.00121.00141.00
* 296.00 98.00 47.00133.00109.00 49.00 49.00 44.00 47.00 30.00 81.00 79.00 21.00 75.00 60.00 63.00
* 329.00119.00 51.00154.00117.00 47.00 46.00 45.00 52.00 34.00 91.00 87.00 24.00 83.00 62.00 [illegible] 53.00133.00147.00
* 317.00106.00 50.00158.00118.00 50.00 49.00 45.00 50.00 34.00 89.00 85.00 25.00 75.00 65.00 67.
* 325.00111.00 53.00145.00120.00 49.00 47.00 42.00 52.00 34.00 90.00 89.00 23.00 74.00 65.00 [illegible] 58.00131
* 293.00 94.00 49.00142.00114.00 43.00 45.00 41.00 44.00 30.00 81.00 82.00 23.00 65.00 59.00 [illegible]
* 302.00102.00 50.00146.00114.00 49.00 47.00 41.00 49.00 33.00 82.00 83.00 22.00 81.00 53.00 64. 53.00120.
* 305.00 99.00 49.00150.00115.00 48.00 46.00 43.00 47.00 30.00 84.00 81.00 23.00 73.00 57.00 67.00 54.0011
* 308.00103.00 49.00152.00122.00 46.00 45.00 41.00 47.00 23.00 86.00 83.00 22.00 72.00 59.00 62.00 56.00122
```

123	*	315.	00109.	00	52.	00161.	00120.	00	46.	00	46.	00	41.	00	49.	00	33.	00	89.	00	92.	00	23.	00	75.	00	63.	00	65.	00	56.	00	116.	00	137.	00	26.	0
124	*	29?.	00	98.	00	45.	00	134.	00	110.	00	48.	00	47.	00	42.	00	41.	00	28.	00	76.	00	77.	00	19.	00	68.	00	55.	00	59.	00	44.	00	108.	00	121.	00	26.	0
125	*	291.	00	93.	00	47.	00	150.	00	113.	00	45.	00	46.	00	40.	00	49.	00	30.	00	83.	00	83.	00	22.	00	73.	00	56.	00	60.	00	44.	00	95.	00	105.	00	23.	0
126	*	239.	00	74.	00	40.	00	128.	00	103.	00	47.	00	44.	00	41.	00	41.	00	27.	00	73.	00	75.	00	16.	00	57.	00	49.	00	58.	00	44.	00	95.	00	105.	00	23.	0
127	*	326.	00	111.	00	50.	00	159.	00	121.	00	49.	00	50.	00	46.	00	49.	00	34.	00	88.	00	87.	00	24.	00	81.	00	59.	00	67.	00	57.	00	135.	00	141.	00	27.	0
128	*	316.	00	109.	00	51.	00	156.	00	120.	00	49.	00	48.	00	42.	00	54.	00	34.	00	85.	00	81.	00	23.	00	76.	00	60.	00	66.	00	59.	00	126.	00	143.	00	26.	0
129	*	324.	00	109.	00	50.	00	154.	00	116.	00	48.	00	47.	00	44.	00	49.	00	31.	00	88.	00	93.	00	26.	00	78.	00	61.	00	65.	00	57.	00	124.	00	147.	00	26.	0
130	*	324.	00	111.	00	48.	00	159.	00	121.	00	49.	00	48.	00	42.	00	49.	00	34.	00	90.	00	94.	00	27.	00	76.	00	60.	00	66.	00	54.	00	133.	00	149.	00	26.	0
131	*	296.	00	98.	00	48.	00	146.	00	113.	00	45.	00	44.	00	39.	00	47.	00	32.	00	82.	00	85.	00	22.	00	72.	00	57.	00	58.	00	53.	00	119.	00	139.	00	25.	0
132	*	284.	00	94.	00	49.	00	140.	00	114.	00	48.	00	46.	00	44.	00	45.	00	31.	00	82.	00	84.	00	21.	00	70.	00	57.	00	61.	00	49.	00	115.	00	132.	00	25.	0
133	*	266.	00	84.	00	46.	00	135.	00	113.	00	48.	00	49.	00	46.	00	42.	00	28.	00	79.	00	81.	00	19.	00	65.	00	52.	00	60.	00	46.	00	106.	00	123.	00	26.	0
134	*	324.	00	116.	00	51.	00	152.	00	117.	00	48.	00	47.	00	45.	00	50.	00	32.	00	88.	00	85.	00	27.	00	80.	00	58.	00	64.	00	57.	00	133.	00	146.	00	26.	0
135	*	321.	00	112.	00	49.	00	159.	00	117.	00	47.	00	47.	00	39.	00	50.	00	33.	00	88.	00	86.	00	26.	00	78.	00	63.	00	64.	00	55.	00	132.	00	150.	00	27.	0
136	*	321.	00	109.	00	50.	00	157.	00	119.	00	48.	00	47.	00	42.	00	51.	00	32.	00	86.	00	94.	00	25.	00	80.	00	60.	00	64.	00	57.	00	136.	00	151.	00	27.	0
137	*	320.	00	111.	00	50.	00	154.	00	118.	00	48.	00	48.	00	42.	00	52.	00	32.	00	90.	00	88.	00	25.	00	79.	00	61.	00	64.	00	56.	00	133.	00	144.	00	26.	0
138	*	308.	00	102.	00	51.	00	151.	00	115.	00	49.	00	49.	00	45.	00	47.	00	33.	00	88.	00	87.	00	24.	00	71.	00	62.	00	65.	00	56.	00	132.	00	146.	00	27.	0
139	*	261.	00	79.	00	45.	00	131.	00	106.	00	49.	00	48.	00	46.	00	43.	00	30.	00	74.	00	74.	00	18.	00	65.	00	57.	00	60.	00	43.	00	95.	00	117.	00	25.	0
140	*	299.	00	98.	00	49.	00	150.	00	113.	00	48.	00	49.	00	43.	00	48.	00	32.	00	85.	00	84.	00	23.	00	79.	00	52.	00	64.	00	56.	00	120.	00	141.	00	26.	0
141	*	324.	00	108.	00	53.	00	155.	00	113.	00	47.	00	47.	00	43.	00	53.	00	34.	00	86.	00	83.	00	24.	00	78.	00	60.	00	67.	00	59.	00	124.	00	140.	00	26.	0
142	*	331.	00	110.	00	51.	00	152.	00	118.	00	51.	00	50.	00	42.	00	52.	00	34.	00	84.	00	79.	00	25.	00	82.	00	64.	00	67.	00	56.	00	127.	00	156.	00	28.	0
143	*	321.	00	109.	00	48.	00	147.	00	116.	00	48.	00	47.	00	44.	00	48.	00	33.	00	87.	00	80.	00	23.	00	78.	00	63.	00	64.	00	56.	00	134.	00	143.	00	27.	0
144	*	291.	00	95.	00	46.	00	139.	00	111.	00	48.	00	49.	00	41.	00	45.	00	29.	00	81.	00	79.	00	23.	00	74.	00	50.	00	63.	00	52.	00	116.	00	136.	00	27.	0
145	*	335.	00	116.	00	53.	00	154.	00	114.	00	48.	00	50.	00	40.	00	54.	00	34.	00	92.	00	80.	00	25.	00	82.	00	63.	00	67.	00	59.	00	134.	00	155.	00	27.	0
146	*	271.	00	84.	00	45.	00	131.	00	105.	00	46.	00	47.	00	37.	00	42.	00	26.	00	77.	00	78.	00	20.	00	66.	00	50.	00	55.	00	47.	00	111.	00	132.	00	24.	0
147	*	267.	00	85.	00	46.	00	129.	00	104.	00	47.	00	47.	00	38.	00	43.	00	27.	00	76.	00	80.	00	20.	00	65.	00	52.	00	56.	00	48.	00	113.	00	143.	00	25.	0
148	*	292.	00	93.	00	48.	00	147.	00	116.	00	49.	00	48.	00	46.	00	48.	00	29.	00	82.	00	85.	00	21.	00	76.	00	57.	00	62.	00	52.	00	112.	00	128.	00	28.	0
149	*	318.	00	105.	00	49.	00	149.	00	115.	00	50.	00	49.	00	42.	00	51.	00	30.	00	84.	00	82.	00	26.	00	80.	00	61.	00	61.	00	55.	00	128.	00	138.	00	28.	0
150	*	285.	00	90.	00	47.	00	144.	00	113.	00	49.	00	49.	00	46.	00	44.	00	28.	00	80.	00	83.	00	23.	00	73.	00	53.	00	62.	00	50.	00	109.	00	130.	00	26.	0
151	*	319.	00	103.	00	53.	00	150.	00	119.	00	48.	00	48.	00	45.	00	51.	00	33.	00	88.	00	83.	00	25.	00	78.	00	62.	00	64.	00	57.	00	126.	00	139.	00	28.	0
152	*	280.	00	90.	00	45.	00	134.	00	111.	00	46.	00	45.	00	44.	00	45.	00	29.	00	80.	00	81.	00	22.	00	65.	00	58.	00	59.	00	50.	00	114.	00	139.	00	25.	0
153	*	288.	00	92.	00	45.	00	136.	00	111.	00	47.	00	47.	00	40.	00	48.	00	29.	00	80.	00	80.	00	23.	00	62.	00	65.	00	60.	00	50.	00	120.	00	140.	00	24.	0
154	*	271.	00	88.	00	45.	00	135.	00	109.	00	51.	00	50.	00	44.	00	42.	00	27.	00	78.	00	81.	00	21.	00	70.	00	55.	00	61.	00	45.	00	105.	00	133.	00	26.	0
155	*	298.	00	97.	00	48.	00	149.	00	113.	00	48.	00	48.	00	41.	00	47.	00	28.	00	84.	00	82.	00	22.	00	70.	00	57.	00	61.	00	53.	00	117.	00	140.	00	26.	0
156	*	336.	00	115.	00	48.	00	153.	00	115.	00	48.	00	47.	00	39.	00	51.	00	32.	00	87.	00	82.	00	25.	00	85.	00	66.	00	64.	00	59.	00	135.	00	149.	00	27.	0
157	*	305.	00	102.	00	52.	00	145.	00	117.	00	47.	00	47.	00	40.	00	48.	00	29.	00	84.	00	85.	00	23.	00	76.	00	56.	00	61.	00	56.	00	122.	00	145.	00	26.	0
158	*	285.	00	94.	00	46.	00	145.	00	109.	00	44.	00	43.	00	40.	00	45.	00	29.	00	82.	00	79.	00	20.	00	72.	00	50.	00	57.	00	52.	00	119.	00	134.	00	24.	0
159	*	298.	00	100.	00	50.	00	142.	00	112.	00	49.	00	48.	00	42.	00	48.	00	31.	00	84.	00	81.	00	22.	00	70.	00	57.	00	61.	00	55.	00	122.	00	143.	00	26.	0
160	*	327.	00	108.	00	53.	00	155.	00	119.	00	50.	00	50.	00	43.	00	50.	00	35.	00	90.	00	84.	00	25.	00	77.	00	64.	00	69.	00	58.	00	136.	00	164.	00	28.	0
161	*	316.	00	107.	00	48.	00	151.	00	115.	00	48.	00	49.	00	46.	00	49.	00	32.	00	86.	00	84.	00	23.	00	74.	00	62.	00	63.	00	55.	00	126.	00	141.	00	25.	0
162	*	299.	00	94.	00	50.	00	150.	00	119.	00	49.	00	48.	00	42.	00	48.	00	32.	00	81.	00	76.	00	21.	00	68.	00	65.	00	63.	00	55.	00	124.	00	138.	00	27.	0
163	*	296.	00	99.	00	49.	00	142.	00	111.	00	45.	00	47.	00	44.	00	48.	00	32.	00	83.	00	83.	00	22.	00	75.	00	55.	00	63.	00	53.	00	120.	00	133.	00	26.	0
164	*	297.	00	93.	00	49.	00	148.	00	114.	00	49.	00	50.	00	44.	00	50.	00	33.	00	90.	00	79.	00	22.	00	70.	00	54.	00	62.	00	52.	00	117.	00	159.	00	26.	0
165	*	298.	00	94.	00	50.	00	145.	00	115.	00	49.	00	48.	00	43.	00	48.	00	29.	00	84.	00	83.	00	22.	00	70.	00	60.	00	61.	00	50.	00	116.	00	139.	00	26.	0
166	*	304.	00	101.	00	48.	00	146.	00	113.	00	49.	00	47.	00	42.	00	50.	00	31.	00	84.	00	84.	00	23.	00	71.	00	65.	00	62.	00	59.	00	120.	00	141.	00	26.	0
167	*	302.	00	98.	00	51.	00	144.	00	111.	00	45.	00	46.	00	43.	00	48.	00	32.	00	83.	00	81.	00	24.	00	71.	00	60.	00	63.	00	55.	00	118.	00	141.	00	26.	0
168	*	307.	00	103.	00	50.	00	144.	00	110.	00	48.	00	49.	00	44.	00	52.	00	33.	00	82.	00	79.	00	23.	00	75.	00	59.	00	64.	00	53.	00	128.	00	144.	00	25.	0
169	*	297.	00	97.	00	49.	00	145.	00	115.	00	50.	00	49.	00	42.	00	47.	00	31.	00	83.	00	83.	00	23.	00	75.	00	57.	00	64.	00	52.	00	117.	00	137.	00	26.	0
170	*	309.	00	102.	00	50.	00	153.	00	118.	00	49.	00	49.	00	43.	00	48.	00	33.	00	83.	00	81.	00	25.	00	75.	00	57.	00	65.	00	54.	00	124.	00	144.	00	26.	0
171	*	293.	00	97.	00	53.	00	150.	00	119.	00	48.	00	49.	00	42.	00	52.	00	33.	00	85.	00	81.	00	23.	00	69.	00	58.	00	63.	00	54.	00	112.	00	137.	00	24.	0
172	*	295.	00	98.	00	44.	00	145.	00	113.	00	48.	00	47.	00	44.	00	47.	00	30.	00	83.	00	82.	00	23.	00	69.	00	59.	00	62.	00	54.	00	121.	00	138.	00	24.	0
173	*	286.	00	92.	00	44.	00	141.	00	112.	00	47.	00	45.	00	40.	00	46.	00	30.	00	82.	00	83.	00	20.	00	72.	00	51.	00	61.	00	52.	00	120.	00	135.	00	24.	0
174	*	281.	00	89.	00	49.	00	139.	00	112.	00	47.	00	47.	00	43.	00	45.	00	29.	00	81.	00	81.	00	22.	00	67.	00	59.	00	62.	00	49.	00	116.	00	133.	00	26.	0
175	*	295.	00	97.	00	49.	00	144.	00	114.	00	49.	00	48.	00	46.	00	47.	00	31.	00	83.	00	80.	00	22.	00	66.	00	59.	00	64.	00	51.	00	117.	00	141.	00	26.	0
176	*	323.	00	114.	00	52.	00	155.	00	119.	00	45.	00	44.	00	43.	00	51.	00	34.	00	88.	00	84.	00	24.	00	79.	00	60.	00	67.	00	58.	00	125.	00	138.	00	24.	0
177	*	255.	00	85.	00	44.	00	129.	00	107.	00	48.	00	47.	00	41.	00	41.	00	25.	00	75.	00	78.	00	18.	00	62.	00	50.	00	55.	00	45.	00	101.	00	128.	00	24.	0
178	*	293.	00	95.	00	47.	00	144.	00	108.	00	47.	00	48.	00	44.	00	47.	00	31.	00	83.	00	82.	00	22.	00	69.	00	57.	00	65.	00	51.	00	113.	00	129.	00	23.	0
179	*	272.	00	87.	00	47.	00	133.	00	110.	00	51.	00	50.	00	45.	00	44.	00	29.	00	77.	00	81.	00	19.	00	66.	00	52.	00	61.	00	48.	00	102.	00	130.	00	26.	0
180	*	276.	00	92.	00	46.	00	143.	00	113.	00	45.	00	44.	00	45.	00	46.	00	30.	00	85.	00	82.	00	19.	00	66.	00	55.	00	60.	00	49.	00	112.	00	130.	00	26.	0
181	*	307.	00	100.	00	49.	00	150.	00	114.	00	48.	00	48.	00	43.	00	41.	00	49.	00	30.	00	89.	00	87.	00	23.	00	70.	00	57.	00	60.	00	52.	00	124.	00	129.	00	25.	0
182	*	334.	00	111.	00	52.	00	160.	00	121.	00	50.	00	49.	00	40.	00	54.	00	32.	00	91.	00	88.	00	23.	00	81.	00	61.	00	66.	00	53.	00	140.	00	162.	00	24.	0
183	*	328.	00	112.	00	53.	00	154.	00	121.	00	46.	00	46.	00	42.	00	50.	00	31.	00	94.	00	94.	00	25.	00	73.	00	65.	00	63.	00	58.	00	129.	00	130.	00	27.	0
184	*	333.	00	114.	00	49.	00	156.	00	119.	00	47.	00	47.	00	37.	00	48.	00	31.	00	92.	00	91.	00	25.	00	79.	00	63.	00	65.	00	61.	00	132.	00	146.	00	21.	0
185	*	265.	00	84.	00	48.	00	141.	00	111.	00	47.	00	47.	00	42.	00	44.	00	26.	00	81.	00	83.	00	19.	00	63.	00	50.	00	58.	00	46.	00	106.	00	121.	00	25.	0
186	*	324.	00	109.	00	51.	00	160.	00	114.	00	48.	00	47.	00	40.	00	50.	00	32.	00	89.	00	86.	00	26.	00	84.	00	56.	00	64.	00	50.	00	129.	00	150.	00	27.	0
187	*	304.	00	102.	00	50.	00	145.	00	113.	00	46.	00	45.	00	44.	00	49.	00	28.	00	86.	00	85.	00	22.	00	69.	00	59.	00	61.	00	55.	00	122.	00	150.	00	27.	0
188	*	312.	00	105.	00	50.	00	148.	00	116.	00	44.	00	47.	00	40.	00	51.	00	32.	00	85.	00	87.	00	23.	00	72.	00	62.	00	64.	00	55.	00	127.	00	152.	00	27.	0

*	307.00104.00	48.00149.00110.00	49.00	47.00	40.00	51.00	32.00	87.00	86.00	23.00	74.00	55.00	64.00	56.00121.00132.00					26.00
*	300.00 99.00	47.00147.00112.00	45.00	45.00	43.00	47.00	30.00	84.00	84.00	23.00	76.00	54.00	64.00	53.00126.00131.00					25.00
*	249.00 79.00	44.00125.00107.00	47.00	47.00	42.00	41.00	25.00	78.00	83.00	18.00	56.00	52.00	57.00	41.00	97.00112.00				24.00
*	337.00101.00	45.00150.00113.00	51.00	49.00	44.00	52.00	30.00	86.00	86.00	24.00	77.00	59.00	63.00	57.00123.00160.00					26.00
*	305.00100.00	50.00153.00120.00	49.00	49.00	45.00	50.00	32.00	83.00	83.00	24.00	67.00	65.00	65.00	56.00126.00134.00					26.00
*	285.00 96.00	50.00141.00111.00	45.00	44.00	41.00	46.00	30.00	84.00	81.00	22.00	64.00	57.00	60.00	52.00115.00144.00					25.00
*	327.00113.00	55.00163.00121.00	49.00	49.00	42.00	54.00	34.00	91.00	88.00	26.00	80.00	59.00	68.00	64.00140.00127.00					27.00
*	321.00114.00	54.00150.00113.00	45.00	46.00	41.00	49.00	33.00	86.00	81.00	22.00	79.00	54.00	63.00	58.00131.00142.00					26.00
*	347.00117.00	53.00170.00118.00	49.00	49.00	37.00	56.00	31.00	98.00	92.00	28.00	83.00	66.00	64.00	64.00142.00166.00					27.00
*	288.00 94.00	46.00143.00113.00	46.00	45.00	41.00	49.00	30.00	84.00	81.00	22.00	68.00	60.00	60.00	51.00108.00140.00					25.00
*	311.00104.00	51.00154.00114.00	49.00	49.00	40.00	50.00	30.00	90.00	85.00	23.00	72.00	60.00	62.00	59.00127.00155.00					27.00
*	334.00113.00	52.00153.00117.00	51.00	51.00	41.00	55.00	30.00	93.00	86.00	24.00	73.00	70.00	64.00	61.00134.00156.00					26.00
*	311.00104.00	47.00154.00116.00	46.00	46.00	40.00	49.00	30.00	88.00	89.00	21.00	71.00	55.00	61.00	58.00128.00152.00					26.00
*	317.00103.00	49.00153.00117.00	50.00	49.00	46.00	50.00	30.00	89.00	88.00	23.00	70.00	61.00	64.00	59.00127.00159.00					27.00
*	327.00102.00	48.00154.00120.00	52.00	51.00	44.00	51.00	33.00	90.00	90.00	24.00	80.00	59.00	67.00	58.00131.00142.00					27.00
*	220.00 84.00	43.00117.00102.00	49.00	48.00	45.00	37.00	27.00	72.00	75.00	14.00	55.00	48.00	58.00	40.00	78.00	91.00			22.00
*	299.00 99.00	50.00145.00111.00	51.00	50.00	42.00	48.00	29.00	86.00	82.00	22.00	67.00	62.00	61.00	52.00115.00122.00					25.00
*	272.00 99.00	47.00143.00113.00	46.00	47.00	41.00	44.00	28.00	82.00	84.00	21.00	65.00	55.00	58.00	50.00110.00130.00					24.00
*	269.00 86.00	48.00134.00109.00	50.00	49.00	45.00	43.00	29.00	79.00	76.00	21.00	67.00	53.00	60.00	47.00100.00128.00					24.00
*	280.00 90.00	48.00138.00109.00	49.00	47.00	43.00	49.00	26.00	82.00	87.00	21.00	69.00	53.00	61.00	48.00111.00129.00					25.00
*	286.00 90.00	50.00143.00113.00	49.00	48.00	44.00	50.00	28.00	87.00	83.00	22.00	68.00	60.00	63.00	50.00112.00137.00					26.00
*	302.00 94.00	49.00150.00115.00	49.00	46.00	45.00	47.00	33.00	87.00	87.00	21.00	70.00	56.00	65.00	53.00124.00125.00					26.00
*	302.00100.00	50.00151.00111.00	49.00	48.00	42.00	49.00	31.00	94.00	82.00	23.00	73.00	56.00	66.00	56.00127.00144.00					26.00
*	305.00102.00	49.00149.00114.00	46.00	45.00	38.00	49.00	29.00	85.00	88.00	22.00	71.00	60.00	60.00	55.00121.00128.00					25.00
*	209.00 69.00	45.00119.00 92.00	49.00	48.00	48.00	35.00	24.00	67.00	74.00	14.00	51.00	48.00	54.00	39.00	67.00	74.00			22.00
*	303.00103.00	47.00151.00112.00	48.00	46.00	44.00	51.00	32.00	85.00	83.00	23.00	70.00	59.00	65.00	54.00121.00146.00					25.00
*	259.00 98.00	50.00146.00111.00	47.00	47.00	43.00	50.00	31.00	83.00	77.00	23.00	69.00	58.00	63.00	55.00115.00133.00					21.00
*	314.00105.00	50.00149.00113.00	48.00	46.00	42.00	52.00	31.00	84.00	80.00	21.00	72.00	60.00	63.00	57.00121.00138.00					26.00
*	279.00 93.00	49.00140.00109.00	50.00	49.00	44.00	47.00	30.00	83.00	81.00	20.00	62.00	60.00	62.00	50.00106.00126.00					25.00
*	284.00 93.00	48.00140.00108.00	49.00	49.00	44.00	48.00	31.00	81.00	80.00	23.00	66.00	57.00	63.00	51.00113.00138.00					26.00
*	244.00 79.00	43.00125.00101.00	45.00	44.00	42.00	41.00	26.00	74.00	75.00	18.00	63.00	53.00	55.00	43.00	88.00110.00				23.00
*	273.00 89.00	45.00142.00107.00	46.00	48.00	40.00	47.00	29.00	80.00	80.00	22.00	63.00	56.00	59.00	50.00109.00131.00					25.00
*	291.00 95.00	46.00142.00111.00	46.00	47.00	40.00	45.00	29.00	85.00	80.00	22.00	73.00	56.00	59.00	56.00113.00136.00					24.00
*	279.00 98.00	45.00143.00112.00	48.00	49.00	43.00	44.00	28.00	82.00	84.00	22.00	65.00	52.00	62.00	49.00141.00115.00					24.00
*	285.00 90.00	48.00142.00114.00	47.00	49.00	42.00	43.00	27.00	83.00	82.00	21.00	68.00	58.00	58.00	52.00116.00149.00					24.00
*	303.00101.00	48.00145.00108.00	49.00	48.00	42.00	48.00	30.00	82.00	80.00	22.00	70.00	59.00	63.00	55.00126.00147.00					24.00
*	301.00 97.00	48.00149.00111.00	51.00	50.00	41.00	50.00	28.00	92.00	88.00	23.00	72.00	57.00	64.00	52.00123.00155.00					26.00
*	315.00106.00	48.00158.00122.00	46.00	48.00	38.00	52.00	33.00	86.00	84.00	25.00	67.00	63.00	66.00	57.00130.00156.00					26.00
*	265.00 84.00	45.00124.00110.00	51.00	50.00	42.00	45.00	28.00	80.00	82.00	20.00	63.00	56.00	61.00	46.00115.00121.00					26.00
*	273.00 85.00	49.00134.00111.00	50.00	49.00	46.00	44.00	27.00	82.00	80.00	20.00	63.00	58.00	60.00	50.00115.00134.00					27.00
*	324.00109.00	50.00149.00114.00	47.00	47.00	42.00	51.00	32.00	85.00	85.00	23.00	77.00	62.00	63.00	62.00125.00137.00					27.00
*	318.00101.00	51.00154.00120.00	53.00	52.00	43.00	50.00	30.00	87.00	87.00	24.00	73.00	62.00	63.00	59.00134.00150.00					26.00
*	306.00 99.00	48.00143.00112.00	50.00	48.00	43.00	49.00	31.00	85.00	82.00	24.00	71.00	58.00	63.00	55.00120.00155.00					26.00
*	286.00 88.00	48.00135.00105.00	47.00	49.00	39.00	45.00	29.00	78.00	78.00	21.00	67.00	56.00	61.00	54.00109.00157.00					25.00
*	329.00107.00	52.00157.00116.00	49.00	48.00	43.00	52.00	33.00	87.00	87.00	24.00	73.00	66.00	66.00	61.00143.00151.00					26.00
*	302.00 95.00	50.00145.00117.00	48.00	47.00	46.00	49.00	30.00	82.00	82.00	23.00	70.00	59.00	61.00	54.00135.00138.00					27.00
*	325.00107.00	49.00155.00115.00	49.00	49.00	43.00	54.00	33.00	84.00	83.00	24.00	81.00	61.00	64.00	60.00136.00145.00					26.00
*	294.00 94.00	47.00144.00102.00	47.00	45.00	38.00	44.00	27.00	78.00	79.00	26.00	72.00	54.00	59.00	52.00127.00140.00					24.00
*	264.00 81.00	46.00134.00110.00	47.00	46.00	41.00	42.00	29.00	76.00	78.00	21.00	66.00	50.00	58.00	46.00115.00135.00					23.00
*	315.00102.00	48.00144.00109.00	50.00	48.00	44.00	48.00	30.00	85.00	83.00	23.00	78.00	58.00	64.00	60.00115.00154.00					27.00
*	286.00 86.00	48.00138.00111.00	47.00	45.00	40.00	44.00	29.00	81.00	80.00	22.00	63.00	55.00	58.00	54.00114.00138.00					23.00
*	275.00 89.00	47.00140.00110.00	45.00	45.00	39.00	44.00	30.00	80.00	81.00	21.00	66.00	51.00	58.00	46.00117.00140.00					25.00
*	289.00 91.00	52.00145.00116.00	50.00	48.00	47.00	45.00	32.00	83.00	84.00	22.00	67.00	57.00	63.00	51.00114.00134.00					26.00
*	295.00 96.00	47.00149.00114.00	48.00	49.00	43.00	45.00	30.00	84.00	84.00	22.00	75.00	55.00	64.00	52.00114.00135.00					26.00
*	296.00 95.00	49.00151.00118.00	48.00	47.00	46.00	45.00	32.00	84.00	86.00	22.00	73.00	58.00	64.00	54.00123.00132.00					26.00
*	329.00104.00	51.00159.00114.00	50.00	50.00	45.00	56.00	34.00	89.00	90.00	24.00	76.00	59.00	66.00	56.00	151.00				27.00
*	317.00112.00	53.00158.00119.00	49.00	47.00	42.00	53.00	34.00	90.00	89.00	24.00	78.00	59.00	64.00	56.00120.00146.00					27.00
*	283.00 90.00	49.00135.00109.00	48.00	46.00	44.00	44.00	30.00	82.00	83.00	21.00	68.00	55.00	61.00	49.00110.00140.00					27.00
*	310.00101.00	48.00149.00116.00	50.00	50.00	45.00	49.00	32.00	84.00	79.00	24.00	72.00	63.00	65.00	55.00124.00151.00					27.00
*	284.00 94.00	49.00133.00112.00	48.00	47.00	47.00	43.00	28.00	82.00	85.00	18.00	63.00	53.00	59.00	48.00	95.00104.00				25.00
*	293.00 96.00	51.00143.00114.00	51.00	49.00	41.00	49.00	32.00	83.00	79.00	23.00	65.00	51.00	65.00	53.00115.00133.00					26.00
*	310.00103.00	48.00152.00118.00	50.00	50.00	44.00	52.00	31.00	86.00	88.00	25.00	72.00	61.00	64.00	55.00127.00156.00					26.00
*	267.00 79.00	46.00137.00107.00	45.00	44.00	39.00	42.00	28.00	78.00	77.00	21.00	59.00	50.00	57.00	49.00100.00131.00					22.00
*	291.00 95.00	47.00150.00107.00	46.00	46.00	44.00	41.00	28.00	76.00	79.00	19.00	59.00	53.00	58.00	45.00	91.00129.00				22.00
*	284.00 90.00	49.00145.00112.00	46.00	45.00	39.00	46.00	32.00	82.00	85.00	24.00	70.00	52.00	61.00	52.00111.00139.00					24.00
*	291.00 95.00	46.00147.00115.00	46.00	45.00	42.00	45.00	31.00	81.00	84.00	24.00	70.00	55.00	62.00	52.00	94.00141.00				24.00
*	260.00 84.00	48.00134.00110.00	45.00	43.00	43.00	44.00	29.00	81.00	83.00	22.00	65.00	50.00	60.00	48.00109.00129.00					23.00
*	274.00 88.00	45.00146.00111.00	46.00	46.00	38.00	47.00	30.00	73.00	83.00	22.00	65.00	55.00	58.00	50.00113.00142.00					24.00
*	281.00 92.00	48.00144.00114.00	50.00	49.00	45.00	47.00	31.00	81.00	85.00	21.00	73.00	55.00	62.00	41.00106.00131.00					26.00
*	251.00 75.00	43.00124.00100.00	44.00	44.00	42.00	40.00	25.00	73.00	79.00	18.00	55.00	50.00	54.00	42.00	98.00122.00				23.00
*	295.00 97.00	51.00144.00115.00	45.00	45.00	42.00	48.00	31.00	85.00	84.00	24.00	70.00	54.00	63.00	55.00118.00147.00					25.00
*	274.00 95.00	44.00142.00114.00	46.00	44.00	40.00	43.00	30.00	82.00	81.00	21.00	64.00	57.00	58.00	50.00114.00147.00					24.00
*	294.00 94.00	48.00144.00115.00	46.00	45.00	39.00	46.00	32.00	85.00	85.00	22.00	72.00	53.00	61.00	55.00117.00143.00					25.00
*	274.00 96.00	47.00141.00110.00	44.00	44.00	41.00	44.00	29.00	81.00	80.00	22.00	61.00	58.00	59.00	52.00107.00148.00					24.00
*	270.00 95.00	47.00138.00109.00	45.00	45.00	42.00	47.00	28.00	80.00	79.00	22.00	63.00	58.00	58.00	55.00110.00151.00					25.00
*	268.00 82.00	47.00142.00114.00	46.00	45.00	43.00	45.00	29.00	79.00	81.00	20.00	64.00	52.00	59.00	44.00103.00139.00					24.00
*	303.00 99.00	50.00149.00112.00	46.00	45.00	39.00	48.00	30.00	85.00	83.00	24.00	66.00	58.00	60.00	55.00115.00150.00					24.00

Bibliographie :

- J.P. BENZECRI & collaborateurs : "L'analyse des données" - tome 2 - L'analyse des correspondances - DUNOD 1973.

- B. LEROUX et B. ESCOFIER : "Etude de trois problèmes de stabilité en analyse factorielle" - Juin 1973.

- Ph. CIARLET : Cours d'analyse numérique - Ecole Nationale des Ponts et Chaussées - 1973.

- CHANDLER-DAVIS et W.M. KAHAN : "The rotation of eigenvectors by a perturbation" - SIAM J. Numer. Anal. - Vol.7, n° 1, March 1970.

- G. HACQUES : "Mathématiques pour l'informatique" - 3. Algorithmique numérique (1) cours. - Armand COLIN 1971.

TABLE DES MATIERES

I. Sommaire. 1
II. Rappel d'analyse des correspondances. 13
III. Construction des algorithmes 16
IV. Etude des algorithmes ITER. 22
V. Etude pour une approximation de rang r. 62
VI. Résultats pour approximation de rang o. 106
VII. Résultats pour une approximation de rang r. 131
VIII. Conclusion. 222

Annexes :

1. Approximation de rang o. 223
2. Approximation de rang r 227
   Notice d'utilisation du programme trou.
3. Tableau des données utilisées 242

Bibliographie. 250